DATA REDUCTION

LAPORAN PENELITIAN
DIK RUTIN UNIVERSITAS AIRLANGGA
TAHUN ANGGARAN 2004

REDUKSI DATA DENGAN METODE SQUASHING

Peneliti:

Drs. Eto Wuryanto, DEA
Ir. Dyah Herawatie, M.Si.
Rimuljo Hendradi, S.Si., M.Si.

LEMBAGA PENELITIAN UNIVERSITAS AIRLANGGA

Dibiayai Oleh Dana DIK Rutin Universitas Airlangga Tahun 2004
Surat Keputusan Rektor Unair Nomor 4222/J03/PG/2004
Tanggal 1 Juni 2004
Nomor Urut: 14

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS AIRLANGGA

Oktober, 2004

DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS AIRLANGGA
LEMBAGA PENELITIAN DAN
PENGABDIAN KEPADA MASYARAKAT
Kampus C Unair, Jl. Mulyorejo Surabaya 60115 Telp. (031) 5995246, 5995248, 5995247 Fax. (031) 5962066
E-mail : infolemlit@unair.ac.id - http: //lppm.unair.ac.id

IDENTITAS DAN PENGESAHAN
LAPORAN AKHIR HASIL PENELITIAN

1. Judul Penelitian	:	**Reduksi Data dengan Metode Squashing**
a. Macam Penelitian	:	☐ **Fundamental** ☐ **Terapan** ☐ **Pengembangan**
b. Kategori Penelitian	:	☐ **I** ☐ **II** ☐ **III**
2. Kepala Poyek Penelitian		
a. Nama lengkap dan Gelar	:	**Drs. Eto Wuryanto, DEA**
b. Jenis kelamin	:	**Laki-laki**
c. Pangkat/Golongan dan NIP	:	**Penata (Gol. III/c) 131933015**
d. Jabatan Sekarang	:	**Staf Pengajar**
e. Fakultas/Puslit/Jurusan	:	**Fakultas MIPA/Matematika**
f. Univ/Ins./Akademi	:	**Universitas Airlangga**
g. Bidang Ilmu yang diteliti	:	
3. Jumlah Tim Peneliti	:	**3 (tiga) orang**
4. Lokasi Penelitian	:	**Lab Komputer Jurusan Matematika FMIPA Unair**
5. Kerjasama dengan Instansi lain		
a. Nama Instansi	:	-
b. Alamat	:	-
6. Jangka waktu penelitian	:	**6 (enam) bulan**
7. Biaya yang diperlukan	:	**Rp. 3.500.000,00**
8. Hasil Penelitian		**( ) Baik Sekali** **(V) Baik**
		( ) Sedang **( ) Kurang**

Surabaya, 26 Agustus 2005

Mengetahui/Mengesahkan
a.n. Rektor
Lembaga Penelitian Dan Pengabdian Kepada Masyarakat
Ketua,

Prof. Dr. H. Sarmanu, M.S.
NIP 130 701 125

RINGKASAN

REDUKSI DATA DENGAN METODE *SQUASHING*, Eto Wuryanto, Dyah Herawatie dan Rimuljo Hendradi, 2004, 41 halaman.

Pembuatan sub-sampel dengan menggunakan metode sampling acak dan metode sampling acak stratifikasi sebenarnya mudah dilakukan tetapi mempunyai kelemahan. Sub-sampel yang dihasilkan metode sampling acak mempunyai varians yang besar jika sampel induknya heterogen. Sedangkan pada metode sampling acak stratifikasi juga mempunyai kelemahan jika *strata* tidak didefinisikan secara jelas. Selanjutnya ternyata ada metode baru yaitu metode *Squashing*,, permasalahannya adalah bagaimana cara mendapatkan sub-sampel dari metode ini (data *Squashing*)? atau dengan kata lain bagaimana mereduksi data berukuran besar dengan menggunakan metode *Squashing* ? dan selanjutnya bagaimana menerapkan metode *Squashing* tersebut untuk mereduksi data yang berdistribusi Pareto tergeneralisir dengan bantuan program S-Plus ?

Penelitian ini bertujuan untuk mereduksi data berukuran besar dengan metode *squashing* dan menerapkannya untuk mereduksi data yang berdistribusi Pareto tergeneralisir dengan bantuan program S-Plus. Sehingga dapat memberikan solusi alternatif dalam mereduksi data, khususnya untuk data berukuran besar yang berdistribusi Pareto tergeneralisir.

Reduksi data dengan menggunakan metode *squashing* dilakukan dengan cara. Pertama, melakukan pengelompokkan terhadap data induk ke dalam beberapa partisi. Kedua, menentukan pembobot dan *pseudo point* pada setiap kelompok atau partisi. Ketiga, menentukan nilai data *squashing* yang merupakan perkalian antara pembobot dan *pseudo point*, kemudian menghitung jumlahan data induk dan data hasil *squashing*. Dalam penelitian ini, data yang dipakai adalah data bangkitan yang diperoleh dengan cara membangkitkan bilangan acak berdistribusi U(0,1) yang kemudian ditransformasi ke dalam distribusi Pareto tergeneralisir.

Algoritma reduksi data dengan menggunakan metode squashing pada data berukuran besar yang berdistribusi Pareto tegeneralisir diperoleh dengan cara sebagai

berikut : Pertama, dilakukan pengelompokan data ke dalam beberapa kelompok dengan anggota yang sama atau berbeda. Kedua, untuk per kelompok dihitung jumlah anggotanya yang berfungsi sebagai nilai pembobot dan nilai *pseudo point*-nya sama dengan rata-rata atau varians per kelompok.. Ketiga, menentukan perkalian antara jumlah anggota per kelompok dengan rata-rata atau variansnya untuk mendapatkan data *squashing.* Hasil reduksi data dengan menggunakan metode *squashing* diindikasikan dengan adanya persamaan antara jumlahan data hasil *squashing* dan data induknya.

Implementasi algoritma metode dilakukan dengan menggunakan program S-Plus. Sedangkan data yang digunakan pada penelitian ini adalah data bangkitan berdistribusi pareto tergeneralisir dengan $\alpha = 0.5$, $\beta = 0.25$, dan n = 100.000. Penerapan program terhadap data bangkitan dengan 3 jenis pengelompokkan menghasilkan semakin banyak jumlah kelompoknya maka momen sub sampel dari hasil *squashing* mendekati data induknya.

Adapun saran yang ingin disampaikan adalah : Pertama, penerapan metode *squashing* masih menggunakan data bangkitan, diharapkan dapat dicoba pada data sekunder. Kedua, program-program yang terdapat pada penelitian ini tidak hanya dapat diterapkan pada data yang berdistribusi Pareto tergeneralisir tetapi juga memungkinkan untuk data yang berdistribusi lainnya. Ketiga, untuk penelitian lebih lanjut dapat dilakukankan antara lain : penerapan metode *squashing* untuk variabel lebih dari satu, pengkajian terhadap indikator apa yang dapat dipakai untuk menyatakan bahwa sampel induk dapat diwakili oleh sub-sampel yang dihasilkan oleh metode *squashing* misalnya membandingkan suatu estimator suatu parameter yang dihasilkan oleh sampel induk dengan yang dihasilkan oleh sub-sampelnya.

Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Airlangga, Nomor Kontrak : 890/JO3.2/PG/2004, Tanggal : 12 Juli 2004.

SUMMARY

REDUCTION DATA USING SQUASHING METHODS, Eto Wuryanto, Dyah Herawatie dan Rimuljo Hendradi, 2004, 41 pages.

A sub sample of random sampling methods and stratified random sampling methods is actually simple but they have any disadvantages. The sub sample of random sampling methods has a large variance if the main sample is heterogeneous. And the one of stratified random sampling methods also possess a weakness if the strata are not well-defined. In the present, there is a new sampling method that is called Squashing methods. The problem is what steps can be used to obtain a sub sample of Squashing methods? Or how can we reduce a large data to a considerably smaller one by using Squashing methods? And next, may we applicate the S-Plus program of Squashing methods to reduce a data from generalized Pareto distribution?

The goal of this research is to reduce a large data by using Squashing methods and to applicate the methods that has been translated in S-plus program to a data from generalized Pareto distribution. So the research can give an alternative solution in data reduction, especially for a large data from generalized Pareto distribution.

The stages of data reduction using Squashing methods : The first, cluster a main data into any partitions. The second, compute a weighted value and an appropriate pseudo point for each partition. The third, calculate multiplication between the weighted value and the pseudo point to get data squashing so we are able to know that the sum of main data and the one of squashing data have a same value. In this research, we utilize a generated data that is produced by transforming a random number of U(0,1) distribution to a generalized Pareto distribution number.

The algorithm of data reduction using squashing methods for a large data from generalized Pareto distribution is : The first, make any group with equal or unequal elements. The second, each group determine a number of elements as weigted value and its momen (mean or variance) value as pseudo point. The third, compute squashing data that is equal to a number of elements and its pseudo point is

a mean or variance value for each group. Finnally, both the the sum of main data and the one of squashing data have a same value.

Implementation of the squashing's algorithm is done by writing it to S-Plus program. The data that is taken in this research is generated data from generalized Pareto distribution with $\alpha = 0.5$, $\beta = 0.25$, and n = 100.000. Application of the program to the data for three kinds of group yield increasing of number of group cause the momen's value of sub sample from squashing methods approach the one of main data.

Any advice can be present as the following : The first, we hope you can find a real data and applicate squashing methods on it. The second, the S-plus program in this research not only can be used on data from generalized Pareto distribution but may for a data from another distribution. The third, this research can be continued by doing any work as the below : you are able to use squashing methods for multivariable data, finding what indicator may be utilised to know that main data can be represented by sub sample from squashing methods, for exemple : compare a estimator of parameter that is calculated by using the main data with the one of sub sample from squashing methods.

Mathematics Department, Faculty of Mathematics and Sciences, Airlangga University, Contract's Number : 890/JO3.2/PG/2004, Date : 12 Juli 2004.

KATA PENGANTAR

Puji syukur yang sedalam–dalamnya penulis panjatkan kehadirat Allah SWT atas karunia dan rahmat-Nya, sehingga penelitian yang berjudul "Reduksi Data Dengan Metode *Squashing*" ini dapat diselesaikan dengan baik..

Pada kesempatan ini, kami ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada :

1. Rektor Universitas Airlangga
2. Ketua Lembaga Penelitian Universitas Airlangga
3. Dekan FMIPA Universitas Airlangga
4. Semua pihak yang telah membantu menyelesaikan penulisan laporan akhir penelitian ini

Akhir kata kami mengharapkan, semoga penelitian ini dapat memberikan sesuatu yang bermanfaat bagi pembaca. Disamping itu penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh karena itu saran dan kritik dari pembaca sangat kami harapkan.

Surabaya,

Penulis

DAFTAR ISI

HALAMAN JUDUL

LEMBAR IDENTITAS DAN PENGESAHAN ii

RINGKASAN DAN SUMMARY iii

KATA PENGANTAR vii

DAFTAR ISI viii

DAFTAR LAMPIRAN x

BAB I PENDAHULUAN

1.1 Latar Belakang 1

1.2 Rumusan Masalah 3

1.3 Batasan Masalah 3

BAB II TINJAUAN PUSTAKA

2.1 Distribusi Pareto Tergeneralisir 4

2.2 Fungsi Distribusi Kumulatif 4

2.3 Metode Kuantil 4

2.4 Fungsi Pembangkit Momen 5

2.5 Pembangkit Variabel Acak 6

2.6 Metode Newton-Raphson 7

2.7 Metode Squashing 8

2.8 S-PLUS 9

BAB III TUJUAN DAN MANFAAT

3.1 Tujuan 13

3.2 Manfaat 13

BAB IV METODE PENELITIAN

BAB V HASIL DAN PEMBAHASAN

5.1 Fungsi Kumulatif Distribusi Pareto Tergeneralisir 15

5.2 *Generate* Distribusi Pareto Tergeneralisir 16

5.3 Metode *Squashing* 17

5.3.1 Pengertian Data Squashing 17

5.3.2 Pengelompokkan Data 20

5.3.3 Penghitungan Momen 21

5.3.4 Membangkitkan *pseudo data points* dan pembobot 22

5.4 Kasus Khusus 28

5.5 Algoritma Program 30

5.6 Implementasi Algoritma ke Program 34

5.7 Data 35

5.8 Analisis Data 36

BAB VI KESIMPULAN DAN SARAN

6.1 Kesimpulan 39

6.2 Saran 40

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR LAMPIRAN

No.	Judul Lampiran
1	Program
2	Ouput Program

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang sangat pesat menyebabkan keperluan terhadap *database* sangatlah mendesak dan akibatnya menghasilkan data *warehouse* (data yang berukuran sangat besar). Data besar tersebut juga dinamakan *massive dataset* yang biasanya memuat ribuan bahkan jutaan pengamatan dan jumlahnya akan mengalami kenaikan seiring dengan bertambahnya waktu. Hal ini disebabkan karena data ini dikumpulkan dan diproses secara *real time*. Contoh *massive dataset* antara lain : data telekomunikasi, astronomi, biologi komputasi dan internet *logging*. Perkembangan ini mengakibatkan pentingnya ekstraksi informasi untuk membantu pembuatan keputusan, perencanaan strategis, dan monitoring system, yang sering disebut dengan *data mining* atau *knowledge discovery in data bases.*

Dengan jumlah data yang sangat besar dan adanya kenaikan jumlah data dalam *massive dataset* maka baik analisis maupun visualisasi data dengan metode-metode tradisional sulit dikerjakan bahkan tidak bisa dilakukan. Sehingga perlu dicarikan cara supaya dapat dilakukan pengolahan data yang berukuran besar tersebut. Cara paling sederhana adalah melakukan pengurangan jumlah data (reduksi data) atau membuat sub-sampel dengan metode sampling tradisional (konvensional), misalkan metode sampling acak sederhana dan stratifikasi. Pembuatan sub-sampel dalam metode sampling acak dilakukan dengan cara pengambilan satu-satu secara acak yang masing-masing anggota sampel induk mempunyai kemungkinan yang sama untuk terambil. Walaupun metode ini mudah dilakukan tetapi mempunyai kelemahan yaitu sub-sampel yang dihasilkan mempunyai varians yang besar jika sampel induknya heterogen. Sedangkan sub-sampel dari metode sampling acak stratifikasi dihasilkan dengan cara sampel induk dibagi kedalam beberapa kelompok independent (*strata*) kemudian dilakukan pengambilan sejumlah tertentu anggota sub-sampel di setiap kelompok secara acak. Seperti pada metode sampling acak sederhana, metode sampling acak stratifikasi ini juga mempunyai kelemahan jika *strata* tidak didefinisikan secara jelas. Sehingga dengan melihat kelemahan-kelemahan pada kedua metode sampling konvensional tersebut masih diragukan apakah data hasil sampling (sub-sampel) tersebut sudah mewakili *massive dataset* yang merupakan data induknya.

Penelitian terbaru yang dilakukan **DuMouchel et al (1999)** mengusulkan suatu pendekatan baru yang disebut metode *squashing*. Metode ini mereduksi *massive dataset* menjadi *dataset* yang lebih kecil dan dapat diolah yang mempresentasikan data induk baik untuk keperluan inferensi maupun prediksi. Ide dari metode squashing ini adalah mendapatkan nilai-nilai pengamatan baru (*pseudo data point*) sedemikian sehingga nilai momen empirik dari *dataset* hasil *squashing* (data *squashing*) sama dengan nilai yang dihasilkan oleh *dataset* induk.

Dalam reduksi data atau penyampelan data, sifat data dengan "*heavy tail*" merupakan hal yang sangat menarik karena sub sampel yang dihasilkan dari jenis data ini diperlukan jumlah yang besar supaya dapat mempresentasikan *dataset* induk. Salah satu contoh data "*heavy tail*" ini adalah data yang berdistribusi Pareto tergeneralisir yang diindikasikan oleh parameter *tail-nya*. Menurut **Trehin** , analisis statistik yang menggunakan distribusi pareto meliputi antara lain : PABX (telpon) yang dikaitkan dengan jumlah ekstensinya, hubungan komputer dengan ukuran memory (MIPS atau terminal yang terinstalasi).

Oleh karena itu berdasarkan kelemahan yang terjadi di metode sampling konvensional dan luasnya penerapan dari distribusi Pareto dan ciri "*heavy tail*" yang dipunyainya serta melihat sifat data *Squashing* yang cukup menjanjikan maka sangatlah menarik untuk dilakukan penelitian yang berkaitan dengan distribusi Pareto dan data *Squashing*. Penelitian ini diharapkan akan diperoleh sub sampel yang dapat diolah dengan metode-metode statistik secara mudah dan menghasilkan akurasi sesuai keinginan dalam proses pengambilan keputusan (*Inference*). Sehingga problem-problem yang ada dalam *data mining* juga bisa diselesaikan dengan mudah dan akurat. Dalam penelitian ini kemungkinan akan ditemukan beberapa penyelesaian dengan pendekatan numerik dan contoh kasus yang memerlukan komputasi yang rumit. Sehingga untuk mempermudah perhitungannya akan dipakai software statistika S-plus.

1.2 Rumusan Masalah

Dari latar belakang yang telah diuraikan, dapat dirumuskan permasalahan sebagai berikut :

1. Bagaimana mereduksi data berukuran besar dengan menggunakan metode Squashing ?
2. Bagaimana menerapkan metode Squashing untuk mereduksi data yang berdistribusi Pareto tergeneralisir dengan bantuan program S-Plus ?

1.3 Batasan Masalah

Pada penelitian ini batasan masalah yang diberikan adalah metode *squashing* untuk satu variabel dengan data yang berukuran besar yang berdistribusi Pareto.

BAB II

TINJAUAN PUSTAKA

2.1 Distribusi Pareto Tergeneralisir

Definisi 2.1 : Definisi fungsi kepadatan peluang (pdf) dari distribusi Pareto tergeneralisir adalah:

$$f(x)=\frac{1}{\beta}\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)-1} \quad , \ x\geq 0 \ , \ \alpha\neq 0 \ \text{ dan } \ \beta>0 \qquad \text{........} \quad (2.1)$$

(Dimakos, 2000)

2.2 Fungsi Distribusi Kumulatif

Definisi 2.2 : Jika variabel acak *X* mempunyai fungsi kepadatan peluang *f(x)*, maka fungsi distribusi kumulatif dari *X* adalah :

1. $F(x)=\sum_{w\leq x} f(w)$, jika *X* diskret .... (2.2)

2. $F(x)=\int_{-\infty}^{x} f(w)\,dw$, jika *X* kontinu ... (2.3)

(Hogg and Craig, 1995)

2.3 Metode Kuantil

Nilai-nilai yang dibawahnya terdapat sejumlah pecahan atau persentase tertentu dari seluruh pengamatan disebut dengan fraktil atau kuantil. Beberapa istilah yang termasuk fraktil atau kuantil adalah persentil, desil dan kuartil. Berikut ini adalah definisi tentang persentil, desil dan kuartil:

Definisi 2.3 : Persentil adalah nilai-nilai yang digunakan untuk membagi segugus pengamatan menjadi 100 bagian yang sama. Nilai-nilai itu dilambangkan dengan $P_1, P_2, ..., P_{99}$ bersifat bahwa 1% dari seluruh data terletak dibawah P_1, 2% terletak dibawah P_2,..., dan 99 % terletak dibawah P_{99}.

Definisi 2.4 : Desil adalah nilai-nilai yang digunakan untuk membagi segugus pengamatan menjadi 10 bagian yang sama. Nilai-nilai itu dilambangkan dengan $D_1, D_2, ..., D_9$ mempunyai sifat bahwa 10 % data jatuh dibawah D_1, 20 % jatuh dibawah D_2,..., dan 90 % jatuh dibawah D_9.

Definisi 2.5 : Kuartil adalah nilai-nilai yang digunakan untuk membagi segugus pengamatan menjadi 4 bagian yang sama. Nilai-nilai itu dilambangkan dengan Q_1 , Q_2 dan Q_3 mempunyai sifat bahwa 25 % data jatuh dibawah Q_1, 50 % data jatuh dibawah Q_2 , dan 75 % data jatuh dibawah Q_3.

(Walpole,1995)

2.4 Fungsi Pembangkit Momen

Definisi 2.6 : Fungsi pembangkit momen dari variabel acak X diberikan oleh $E(e^{tX})$ dan dinyatakan dengan $M_X(t)$. Jadi

$$M_X(t) = E(e^{tX}) \quad .... (2.4)$$

Momen I : $M'_X(0) = E(X)$ .... (2.5)

Momen II : $M''_X(0) = E(X^2)$ .. (2.6)

(Kartiko, 1986)

Definisi 2.7 : Misalkan $X_1, X_2, \ldots, X_n$ menyatakan suatu sampel acak dengan ukuran n dari suatu distribusi yang diberikan, maka rataan sampel dinyatakan dengan statistik :

$$\overline{X} = \frac{X_1 + X_2 + \ldots + X_n}{n} = \frac{\sum_{i=1}^{n} X_i}{n} \quad \ldots. \quad (2.7)$$

Definisi 2.8 : Bila $X_1, X_2, \ldots, X_n$ sampel acak ukuran n, maka *variansi sampel* didefinisikan sebagai statistik :

$$S^2 = \sum_{i=1}^{n} \frac{(X_i - \overline{X})^2}{n} \quad \ldots. \quad (2.8)$$

(Walpole, 1986)

2.5 Pembangkit Variabel Acak

Salah satu metode yang sering digunakan untuk membangkitkan variabel acak yang berdistribusi kontinu adalah *Inverse Transformation Method.*

Preposisi 2.1 : Misalkan U merupakan variabel acak yang berdistribusi uniform (0,1). Untuk suatu fungsi distribusi F, jika variabel acak X didefinisikan :

$$X = F^{-1}(U) \quad (2.9)$$

maka variabel acak X mempunyai fungsi distribusi F [$F^{-1}(u)$ didefinisikan sama untuk setiap nilai x dengan $F(x)=u$]

(Ross, 1997)

2.6 Metode Newton-Raphson

Misalkan $g_1(u_1,u_2,u_3) = 0$, $g_2(u_1,u_2,u_3) = 0$ dan $g_3(u_1,u_2,u_3) = 0$ adalah tiga persamaan dengan u_1,u_2 dan u_3 yang tidak diketahui. Langkah-langkah dalam metode Newton-Raphson, sebagai berikut :

1. Asumsikan $\mathbf{u}^0 = \begin{bmatrix} u_1^{\ 0} \\ u_2^{\ 0} \\ u_3^{\ 0} \end{bmatrix}$ sebagai nilai awal atau solusi perkiraan dari sistem tiga persamaan nonlinier dengan tiga variabel yang tidak diketahui :

$$\begin{cases} g_1(u_1,u_2,u_3)=0 \\ g_2(u_1,u_2,u_3)=0 \\ g_3(u_1,u_2,u_3)=0 \end{cases} \tag{2.10}$$

2. Menentukan matriks jacobian dari tiga kesamaan di persamaan (2.10) sehingga diperoleh :

$$\text{Jacobian } J(\mathbf{u}^k) \qquad \mathbf{h}^k = \qquad -g(\mathbf{u}^k)$$

$$\begin{bmatrix} \frac{\partial g_1(\mathbf{u}^k)}{\partial u_1} & \frac{\partial g_1(\mathbf{u}^k)}{\partial u_2} & \frac{\partial g_1(\mathbf{u}^k)}{\partial u_3} \\ \frac{\partial g_2(\mathbf{u}^k)}{\partial u_1} & \frac{\partial g_2(\mathbf{u}^k)}{\partial u_2} & \frac{\partial g_2(\mathbf{u}^k)}{\partial u_3} \\ \frac{\partial g_3(\mathbf{u}^k)}{\partial u_1} & \frac{\partial g_3(\mathbf{u}^k)}{\partial u_2} & \frac{\partial g_3(\mathbf{u}^k)}{\partial u_3} \end{bmatrix} \begin{bmatrix} h_1^{\ k} \\ h_2^{\ k} \\ h_3^{\ k} \end{bmatrix} = -\begin{bmatrix} g_1(\mathbf{u}^k) \\ g_2(\mathbf{u}^k) \\ g_3(\mathbf{u}^k) \end{bmatrix} \tag{2.11}$$

kemudian menentukan nilai $\mathbf{h}^k$ dengan :

$$\mathbf{h}^k = -[J(\mathbf{u}^k)]^{-1} g(\mathbf{u}^k) \tag{2.12}$$

3. Misal $\mathbf{u}^k = \{ \mathbf{u}_1^k, \mathbf{u}_2^k, \mathbf{u}_3^k \}$, nilai $\mathbf{u}^{k+1}$ dihitung secara iterasi untuk k = 0, 1, 2, ... sehingga didapat barisan iterasi $\mathbf{u}^1$, $\mathbf{u}^2$, $\mathbf{u}^3$,... dengan

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathbf{h}^k \tag{2.13}$$

4. Menghentikan proses iterasi ketika nilai selisih antara nilai **u** pada iterasi ke-k dengan iterasi ke-(k-1) kurang dari akurasi yang dikehendaki.

(Anonim)

2.7 Metode *Squashing*

Metode *squashing* merupakan suatu metode yang mereduksi data (mengurangi jumlah data) yang berukuran besar menjadi data yang lebih kecil sehingga data tersebut dapat lebih mudah untuk diolah dan merepresentasikan data induk baik untuk keperluan penarikan kesimpulan (inferensi) maupun prediksi.

Misalkan data sampel induk dapat dituliskan (A,X) atau $(A_1, A_2, ..., A_C, X_1, X_2, ..., X_Q)$ dengan $A_1, A_2, ..., A_C$ adalah variabel kategori dan $X_1, X_2, ..., X_Q$ adalah variabel kuantitatif.

Jika $A = A_{ic}$ dan $X = X_{ij}$ di mana i = 1,2, …, N

j = 1, 2, …, Q

c = 1, 2, …, C

sehingga data sampel induk tersebut dapat dituliskan dalam bentuk matriks sebagai berikut :

$$A_{N \times C} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1C} \\ A_{21} & A_{22} & \cdots & A_{2C} \\ \cdots & \cdots & \cdots & \cdots \\ A_{N1} & A_{N2} & \cdots & A_{NC} \end{bmatrix} \quad \text{dan} \quad X_{N \times Q} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1Q} \\ X_{21} & X_{22} & \cdots & X_{2Q} \\ \cdots & \cdots & \cdots & \cdots \\ X_{N1} & X_{N2} & \cdots & X_{NQ} \end{bmatrix}$$

Jika data *squashing* mempunyai baris M << N, maka dapat dimisalkan :

$B = B_{ic}$ dan $Y = Y_{ij}$ di mana i = 1,2, …, M

j = 1, 2, …, Q

c = 1, 2, …, C

Jika w_i merupakan pembobot untuk setiap i, di mana $\sum_{i=1}^{M} w_i = N$, maka data *squashing* dapat dituliskan dalam bentuk matriks sebagai berikut :

$$B_{MxC} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1C} \\ B_{21} & B_{22} & \cdots & B_{2C} \\ \cdots & \cdots & \cdots & \cdots \\ B_{M1} & B_{M2} & \cdots & B_{MC} \end{bmatrix} \quad ; \quad Y_{MxQ} = \begin{bmatrix} Y_{11} & Y_{12} & \cdots & Y_{1Q} \\ Y_{21} & Y_{22} & \cdots & Y_{2Q} \\ \cdots & \cdots & \cdots & \cdots \\ Y_{M1} & Y_{M2} & \cdots & Y_{MQ} \end{bmatrix}$$

Teknik *squashing* terdiri dari tiga tahap seperti berikut :

1. Pengelompokkan (menggolongkan) data.

   Mengelompokkan data ke dalam beberapa daerah atau partisi.

2. Kalkulasi (penghitungan) momen.

   Menghitung nilai momen empirik dari seluruh data yang ada di setiap partisi.

3. Pembangkit elemen data squashing.

   Membangkitkan data dari setiap partisi dan pembobotnya sedemikian sehingga nilai momen emprik dari data bangkitan yang terboboti sama dengan nilai momen yang telah dihitung pada tahap ke dua.

(Dumouchel et.al, 1999)

2.8 S – PLUS

S-Plus adalah paket program yang memungkinkan membuat program sendiri walaupun didalamnya sudah tersedia banyak program internal yang siap digunakan. Kelebihan dari paket program ini adalah baik program internal maupun program yang pernah dibuat dapat digunakan sebagai sub program dari program yang akan dibuat.

Berikut penjelasan dari beberapa contoh perintah internal dalam S-Plus yang bisa digunakan:

a. function(…)

   function(…) digunakan untuk menunjukkan fungsi yang akan digunakan dalam program.

   Bentuknya adalan : function(…)

b. length(…)

   length(…) merupakan perintah yang digunakan untuk menentukan banyaknya anggota dalam suatu vektor atau list.

   Bentuknya : length(x) untuk x : vektor atau list.

c. sort(…)

   Untuk mengurutkan data numerik dari terkecil sampai ke yang terbesar dan data alfabetis dari a ke z dalam suatu vektor, dengan *default missing values* tidak diproses.

   Bentuknya : sort(x, partial=NULL, na.last=NA), untuk na.list=TRUE maka *missing values* diletakkan di urutan terakhir dan na.list=FALSE, *missing values* muncul diurutan pertama.

d. matrix(...)

   Untuk membentuk sebuah matrik dengan jumlah baris dan kolom tertentu.

   Bentuknya : matrix(data=NA, nrow= ..., ncol=..., byrow=F, dimnames=NULL), untuk nrow : banyaknya baris, ncol : banyaknya kolom, byrow=FALSE berarti pengisian elemen matrik dimulai dari kolom pertama, kolom kedua dan seterusmya, jika byrow=TRUE maka pengisisan dimulai dari baris pertama, kedua sampai dengan baris terakhir, datanya bisa berupa 0, NA atau vektor.

e. rep(...)

Untuk membentuk sebuah vector yang anggotanya merupakan pengulangan suatu nilai tertentu.

Bentuknya : rep(x, times=<<see below>>, length=<<see below>>), untuk x berupa vektor dan *missing values (NA)* boleh dipakai, times : berapa kali pengulangan nilai x, length : banyaknya elemen yang diinginkan, jika times dan length keduanya digunakan maka times tidak diproses.

f. seq()

Membentuk suatu vektor dengan elemen awal, elemen akhir, jarak antar elemen dan panjang barisan yang ditentukan.

Bnetuknya : seq(from=<<see below>>, to=<<see below>>, by=<<see below>>, length=<<see below>>, along=NULL) , untuk from : nilai awal dari barisan, to : batas atas barisan, by : jarak antara dua nilai yang berurutan dalam barisan, length : banyaknya elemen barisan yang dihasilkan bisa befungsi jika argumen to tidak digunakan.

g. for(...)

Untuk melakukan *looping* sebanyak n kali.

Bentuknya : for(i in x) , untuk x berupa vektor sebarang boleh terurut maupun tidak.

h. sum(…)

Untuk menjumlahkan semua bilangan anggota dari suatu vektor.

Bentuknya : sum(x, na.rm=F), untuk x : vektor numerik, na.rm=TRUE maka *missing values (NA)* dalam vektor x tidak berpengaruh pada perhitungan dan jika na.rm=FALSE maka hasilnya adalah *NA* bila dalam vektor x memuat *NA*.

f. mean(...)

Untuk menghitung rata-rata suatu vektor.

Bentuknya adalah : mean(x) , untuk x : vektor

g. var(...)

Untuk menghitung varians suatu vektor.

Bentuknya adalah : var(x) , untuk x : vektor

(Everitt, 1994)

BAB III

TUJUAN DAN MANFAAT PENELITIAN

3.1 Tujuan

Penelitian ini bertujuan :

1. Mereduksi data berukuran besar dengan metode *squashing*.
2. Menerapkan metode Squashing untuk mereduksi data yang berdistribusi Pareto tergeneralisir dengan bantuan program S-Plus.

3.2 Manfaat

Dari penelitian ini diharapkan dapat :

1. Menambah wawasan tentang metode reduksi data dalam analisis statistik.
2. Membantu pekerjaan-pekerjaan yang terkait dengan *data mining* atau *knowledge discovery in data bases*, misalkan : pembuatan keputusan, perencanaan strategis, dan monitoring system.
3. Hasil dari penelitian ini diharapkan dapat digunakan sebagai acuan untuk penelitian yang memerlukan pengolahan data berukuran besar atau *massive dataset*.
4. Dapat dipakai sebagai dasar untuk penelitian-penelitian berikutnya, misalnya : bagaimana penentuan partisi-partisi optimal dalam metode *squashing*, bagaimana penggunaan metode *squashing* dengan dasar *likelihood*.

BAB IV
METODE PENELITIAN

Dalam penelitian ini akan digunakan metode penelitian sebagai berikut :

1. Mengumpulkan alat/bahan yang diperlukan seperti majalah ilmiah, journal, buletin, dan karya ilmiah yang ada kaitannya dengan metode *squashing* , deret taylor, distribusi pareto tergeneralisir, generate variabel acak, metode Newtom-Raphson.
2. Penyusunan algoritma :
   a. Mengelompokkan data bangkitan dengan menggunakan pendekatan kuantil.
   b. Menentukan pembobot dan *pseudo data point* dari masing-masing kelompok.
   c. Menentukan momen ke I (mean) dan momen ke II (variansi) untuk data *squashing* dari masing-masing kelompok.
   d. Menghitung mean dan varians dari *squashing* data bangkitan tersebut.
   e. Penentuan data bangkitan yang berdistribusi pareto tergeneralisir.
3. Mengimplementasikan algoritma yang telah disusun ke dalam program komputer (dengan software S-plus).
4. Mendapatkan data bangkitan yang berfungsi sebagai *dataset* induk.
5. Menguji coba dan menerapkan program yang telah jadi melalui data bangkitan.
6. Memperbaiki dan merevisi program jika terjadi kesalahan.

BAB V

HASIL DAN PEMBAHASAN

5.1 Fungsi Kumulatif Distribusi Pareto Tergeneralisir

Fungsi kumulatif distribusi Pareto tergeneralisir didapatkan dengan mengintegralkan pdf dari distribusi Pareto tergeneralisir pada persamaan (2.1) dan bentuk pengintegralannya adalah sebagai berikut :

$$F(x) = \int_0^x f(x)dx$$

$$= \int_0^x \frac{1}{\beta}\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)-1} dx$$

$$= \frac{1}{\beta}\int_0^x \left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)-1} \frac{d\left(1+\alpha\frac{x}{\beta}\right)}{\frac{\alpha}{\beta}}$$

$$= \frac{1}{\beta}\frac{\beta}{\alpha}\frac{1}{-\frac{1}{\alpha}-1+1}\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)-1+1}\Bigg]_0^x$$

$$= \frac{1}{\alpha}(-\alpha)\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)}\Bigg]_0^x$$

$$= -\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)}\Bigg]_0^x$$

$$= \quad -\left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)}+1$$

$$= \quad 1 - \left(1+\alpha\frac{x}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)} \tag{5.1}$$

5.2 *Generate* Distribusi Pareto Tergeneralisir

Untuk dapat men-*generate* (membangkitkan) data berdistribusi Pareto tergeneralisir, kita harus membangkitkan data berdistribusi Uniform dan hasilnya kita transformasi ke distribusi Pareto tergeneralisir. Misalkan U_i adalah variabel acak dari distribusi Uniform maka transformasi distribusi Uniform ke distribusi Pareto tergeneralisir menggunakan metode *Inverse Transform Technique* adalah :

$$F(x_i) \quad = U_i \tag{5.2}$$

Dari sub bab 5.1 didapatkan :

$$1\text{-}\left(1+\alpha\frac{x_i}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)} \quad = U_i$$

Akan sama artinya dengan :

$$\left(1+\alpha\frac{x_i}{\beta}\right)^{-\left(\frac{1}{\alpha}\right)} \quad = 1\text{-}U_i$$

Jika kedua dipangkatkan $\left(-\frac{\alpha}{1}\right)$ maka :

$$1+\alpha\frac{x_i}{\beta} \quad = \quad (1-U_i)^{-\alpha}$$

Jika ke dua ruas dikurangkan dengan (1) :

$$\alpha \frac{x_i}{\beta} \qquad = (1-U_i)^{-\alpha} \text{-} 1$$

Selanjutnya jika kedua ruas dikalikan $\left(\frac{\beta}{\alpha}\right)$ maka akan diperoleh :

$$x_i \quad = \frac{\beta}{\alpha}\left((1-U_i)^{-\alpha} - 1\right)$$

Sehingga didapatkan variabel acak yang berdistribusi Pareto tergeneralisir adalah sebagai berikut :

$$x_i \quad = \frac{\beta}{\alpha}\left((1-U_i)^{-\alpha} - 1\right) \tag{5.3}$$

5.3 Metode *Squashing*

Metode *squashing* merupakan suatu metode yang mereduksi data (mengurangi jumlah data) yang berukuran besar menjadi data yang lebih kecil sehingga data tersebut dapat lebih mudah untuk diolah dan mempresentasikan data induk baik untuk keperluan penarikan kesimpulan (inferensi) maupun prediksi.

5.3.1 Pengertian Data *Squashing*

Karena data sampel induk yang berukuran besar dapat dimisalkan (A, X), sehingga modelnya dapat diasumsikan sebagai berikut :

$$f(a_1,...,a_C,x_1,...,x_Q : \theta) = P(A_1 = a_1,...,A_C = a_C, X_1 = x_1,...,X_Q = x_Q \mid \theta) \tag{5.4}$$

dengan θ adalah parameter.

Data *squashing* yang dimisalkan (B, Y, w) , w adalah pembobot , diharapkan dapat memberikan informasi yang sama dengan data sampel induk yang memuat informasi mengenai θ. Artinya kedua model yaitu data sampel induk dan data *squashing* akan mempunyai fungsi *likelihood* yang sama dan persamaan log-*likelihood*-nya adalah sebagai berikut :

$$\sum_{i=1}^{M} w_i \log(f(B_{i1},\ldots,B_{iC},Y_{i1},\ldots,Y_{iQ}:\theta)) = \sum_{i=1}^{N} \log(f(A_{i1},\ldots,A_{iC},X_{i1},\ldots,X_{iQ}:\theta)) \tag{5.5}$$

yang berlaku untuk setiap nilai dari A dan θ. $f(A,X,\theta)$ merupakan *relatively smooth function* dari $(X_1,\ldots,X_Q)$ sehingga $(f(A_1,\ldots,A_C,X_1,\ldots,X_Q;\theta))$ dapat dinyatakan dengan deret Taylor di sekitar titik $x=(x_1,\ldots,x_Q)$. Selanjutnya $\log(X_1,X_2,\ldots,X_Q;\theta)$ di sekitar $x=(x_1,\ldots,x_Q)$ dapat dijabarkan sebagai berikut :

$$\log(X_1,X_2,\ldots,X_Q;\theta) \cong f(x_1,x_2,\ldots,x_Q)+$$

$$(X_1-x_1)\frac{\partial f}{\partial X_1}(x_1,x_2,\ldots,x_Q)+(X_2-x_2)\frac{\partial f}{\partial X_2}(x_1,x_2,\ldots,x_Q)$$

$$+\ldots+(X_Q-x_Q)\frac{\partial f}{\partial X_Q}(x_1,x_2,\ldots,x_Q)+\frac{1}{2!}[(X_1-x_1)^2\frac{\partial^2 f}{\partial X_1^{\,2}}(x_1,x_2,\ldots,x_Q)+$$

$$(X_2-x_2)^2\frac{\partial^2 f}{\partial X_2^{\,2}}(x_1,x_2,\ldots,x_Q)+\ldots+(X_Q-x_Q)^2\frac{\partial^2 f}{\partial X_Q^{\,2}}(x_1,x_2,\ldots,x_Q)+$$

$$2(X_1-x_1)(X_2-x_2)\frac{\partial^2 f}{\partial X_1 \partial X_2}(x_1,x_2,\ldots,x_Q)+\ldots+2(X_1-x_1)(X_Q-x_Q)$$

$$\frac{\partial^2 f}{\partial X_1 \partial X_Q}(x_1,x_2,\ldots,x_Q)+\ldots+2(X_{Q-1}-x_{Q-1})(X_Q-x_Q)\frac{\partial^2 f}{\partial X_{Q-1}\partial X_Q}(x_1,x_2,\ldots x_Q)]$$

$$+\frac{1}{3!}[(X_1-x_1)^3\frac{\partial^3 f}{\partial X_1^{\,3}}(x_1,x_2,\ldots,x_Q)+\ldots+(X_Q-x_Q)^3\frac{\partial^3 f}{\partial X_Q^{\,3}}(x_1,x_2,\ldots,x_Q)+$$

$$3(X_1-x_1)^2(X_2-x_2)\frac{\partial^3 f}{\partial X_1^{\,2} X_2}(x_1,x_2,\ldots,x_Q)+\ldots+3(X_1-x_1)^2(X_Q-x_Q)$$

$$\frac{\partial^3 f}{\partial X_1^{\,2}\partial X_Q}(x_1,x_2,\ldots,x_Q)+\ldots+3(X_{Q-1}-x_{Q-1})^2(X_Q-x_Q)\frac{\partial^3 f}{\partial X_{Q-1}^{\,2}\partial X_Q}$$

$$(x_1,x_2,\ldots,x_Q)]+\ldots+\frac{1}{k!}[(X_1-x_1)^k\frac{\partial^k f}{\partial X_1^{\,k}}+\ldots+(X_Q-x_Q)^k\frac{\partial^k f}{\partial X_Q^{\,k}}(x_1,x_2,\ldots,x_Q)$$

$$+k(X_1-x_1)^{k-1}(X_2-x_2)\frac{\partial^k f}{\partial X_1^{\,k-1}\partial X_2}(x_1,x_2,\ldots,x_Q)+\ldots$$

$$=g_1(X_1-x_1)^0(X_2-x_2)^0\ldots(X_Q-x_Q)^0+g_2(X_1-x_1)^1(X_2-x_2)^0\cdots(X_Q-x_Q)^0+$$

$$g_3(X_1-x_1)^0(X_2-x_2)^1\ldots(X_Q-x_Q)^0+$$

$$g_{Q+1}(X_1-x_1)^0(X_2-x_2)^0\cdots(X_Q-x_Q)^0+\cdots$$

$$\approx\sum_{k=1}^{K}g_k\prod_{j=1}^{Q}(X_j-x_j)^{p_{kj}} \tag{5.6}$$

dari persamaan (5.6) $\log(f(a_1,\ldots,a_C,x_1,\ldots,x_Q;\theta))$ dapat dinyatakan sebagai deret Taylor di sekitar titik $x=(x_1,\ldots,x_Q)$ dan dapat dituliskan :

$$\log(f(a_1,\ldots,a_C,x_1,\ldots,x_Q;\theta))\approx\sum_{k=1}^{K}g_k\prod_{j=1}^{Q}(X_j-x_j)^{p_{kj}} \tag{5.7}$$

Pada persamaan (5.7) terdapat K suku deret Taylor. Koefisien g_k hanya bergantung pada $(A_1,\ldots,A_C)$, θ, x dan tidak bergantung pada $(X_1,\ldots,X_Q)$ dan *power vectors* $(p_{k1},\ldots,p_{kQ})$. Untuk Semua Q vektor nonnegative, nilai *power vectors* memenuhi $\sum_j p_{kj}\le d$, d = *degree of approximation*. Penentuan nilai vektor $(p_{k1},\ldots,p_{kQ})$ dimulai dari nol, satu kemudian dilanjutkan hingga *k*, yaitu $p_1=(0,\ldots,0)$ $p_2=(1,0,\ldots,0)$, … , $p_{Q+1}=(0,\ldots,0,1)$, $p_{Q+2}=(2,0,\ldots,0)$ dan seterusnya.

Misalkan terdapat R daerah dan data set lengkap memuat N_r titik pada daerah r, di mana $\sum_{r=1}^{R} N_r = N$. Selanjutnya, data *squashing* juga dimisalkan memuat M_r titik pada daerah r, sehingga $\sum_{i=1}^{M_r} w_i = N_r$. Dari persamaan (5.5) dan (5.7) diperoleh :

$$\sum_{i=1}^{M_r} w_i \sum_{k=1}^{K} g_k \prod_{j=1}^{Q} \left(Y_{ij} - x_j\right)^{p_{kj}} = \sum_{i=1}^{N_r} \sum_{k=1}^{K} g_k \prod_{j=1}^{Q} \left(X_{ij} - x_j\right)^{p_{kj}} \quad , \quad r = 1,\ldots,R \tag{5.8}$$

jika g_k dihitung secara terpisah maka diperoleh persamaan :

$$\sum_{i=1}^{M_r} w_i \prod_{j=1}^{Q} \left(Y_{ij} - x_j\right)^{p_{kj}} = \sum_{i=1}^{N_r} \prod_{j=1}^{Q} \left(X_{ij} - x_j\right)^{p_{kj}} \quad , \quad r = 1,\ldots,R,\ k = 1,\ldots,K \tag{5.9}$$

(DuMouchel,et.al, 1999)

5.3.2 Pengelompokkan Data

Pengelompokkan data merupakan salah satu tahap yang digunakan dalam teknik squashing. Dalam hal ini data akan dikelompokkan ke dalam beberapa daerah atau partisi, sehingga data akan terbentuk menjadi beberapa kelompok. Dalam metode *squashing* terdapat beberapa cara untuk mengelompokkan data akan tetapi dalam tahap ini digunakan metode kuantil. Kuantil merupakan ukuran-ukuran yang menjelaskan atau menunjukkan lokasi sebagian data relatif terhadap keseluruhan data, dimana nilai-nilai yang ada di bawahnya terdapat sejumlah pecahan atau persentase tertentu dari seluruh pengamatan. Mengingat metode kuantil merupakan metode yang sering digunakan dalam mengelompokkan data oleh karena itu digunakan metode kuantil untuk mempermudah pengerjaan dalam pengelompokkan data untuk teknik *squashing*.

5.3.3 Perhitungan Momen

Setelah data dikelompokkan, langkah selanjutnya adalah membangkitkan data dari setiap partisi dan pembobotnya sedemikian sehingga nilai momen sub sampel dari data bangkitan yang terboboti sama dengan nilai momen induknya. Penghitungan momen sub sampel dari data bangkitan dapat dilakukan dengan menghitung nilai mean dan varians.

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n} \quad \text{dan} \quad \text{var}\,ians = \sum_{i=1}^{n} \frac{(X_i - \overline{X})^2}{n} \tag{5.10}$$

Nilai dari momen tersebut akan digunakan untuk menghitung nilai pengamatan baru (*pseudo data points*). Untuk mencocokkan jumlah dari momen dan jumlah *pseudo data points*, didefinisikan derajat kebebasan (df) sebagai berikut :

$$df = M_r(Q+1)$$

dimana : M_r adalah jumlah dari pembobot dalam suatu daerah

Q adalah jumlah dari *pseudo data points* dalam suatu daerah

df adalah jumlah dari parameter (nilai dari w dan Y) yang mengandung *pseudo data point*

5.3.4 Membangkitkan *pseudo data points* dan pembobot

Dari persamaan (5.9), jika X_{ij} adalah data *centering* dan diasumsikan x_j adalah mean dari X_{ij} sehingga $x_j = 0$, akan didapat persamaan :

$$\sum_{i=1}^{M_r} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} = z_k \,, k = 1,2,\ldots,K \tag{5.11}$$

dimana :

$$z_k = \sum_{i=1}^{N_r} \prod_{j=1}^{Q} X_{ij}^{\,p_{kj}}$$

Persamaan (5.11) merupakan sistem K persamaan dengan $M_r(Q+1)$ tidak diketahui, sehingga didapat konstrain

$$w_i \geq 0 \,, \min_i X_{ij} \leq Y_{ij} \leq \max_i X_{ij} \tag{5.12}$$

Dari persamaan (5.12) tidak digunakan pembobot negatif dan nilai variabel di luar daerah dari variabel data induk, karena diinginkan hasil dari data *squashing* akan sama dengan data induk. Namun, nilai dari Y tidak dibatasi pada nilai yang berhubungan dengan X. Untuk mengetahui nilai dari (Y, w), dengan meminimumkan metode estimasi *least square* dari persamaan :

$$S(Y,w) = \sum_{k=1}^{K} u_k \left(z_k - \sum_{i=1}^{M_r} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 \tag{5.13}$$

Turunan terhadap Y_{ij} (*pseudo data points*) pada persamaan (5.13) adalah sebagai berikut :

$$\frac{\partial S(Y,w)}{\partial Y_{ij}} = \sum_{k=1}^{K} 2u_k \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,pkj} \right)\left(-\frac{pkj}{Y_{ij}} w_i \prod_{j=1}^{Q} Y_{ij}^{\,pkj} \right) \tag{5.14}$$

Misalkan :

$$\frac{\partial S(Y,w)}{\partial Y_{ij}} = R_{ij}$$

Turunan R_{ij} terhadap Y_{ij} (*pseudo data point*) pada persamaan (5.14) adalah sebagai berikut :

$$\frac{\partial R_{ij}}{\partial Y_{ij}} = \sum_{k=1}^{K} 2u_k \left[\left(\frac{p_{kj} p_{kj}}{Y_{ij} Y_{ij}} \left(\sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2 \right) + \left(\frac{p_{kj} p_{kj}}{Y_{ij} Y_{ij}} \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right) \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ pkj} \right) \right] \tag{5.15}$$

Turunan R_{ij} terhadap w_i (pembobot) pada persamaan (5.14) adalah sebagai berikut :

$$\frac{\partial R_{ij}}{\partial w_i} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kj}}{Y_{ij}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2 + \frac{p_{kj}}{Y_{ij}} \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ pkj} \right) \right] \tag{5.16}$$

Dan turunan terhadap w_i (pembobot) pada persamaan (5.13) adalah sebagai berikut :

$$\frac{\partial S(Y,w)}{\partial w_i} = \sum_{k=1}^{K} 2u_k \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ pkj} \right) \left(- \prod_{j=1}^{Q} Y_{ij}^{\ pkj} \right) \tag{5.17}$$

Misalkan :

$$\frac{\partial S(Y,w)}{\partial w_i} = P_i$$

Turunan P_i terhadap Y_{ij} (*pseudo data point*) pada persamaan (5.17) adalah sebagai berikut :

$$\frac{\partial P_i}{\partial Y_{ij}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kj}}{Y_{ij}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2 + \frac{p_{kj}}{Y_{ij}} \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ pkj} \right) \right] \tag{5.18}$$

Turunan P_i terhadap w_i (pembobot) pada persamaan (5.18) adalah sebagai berikut :

$$\frac{\partial P_i}{\partial w_i} = \sum_{k=1}^{K} 2u_k \left(\prod_{j=1}^{Q} Y_{ij}^{\;Pkj} \right)^2 \qquad (5.19)$$

Karena dari persamaan (5.15) dan (5.18) setelah disama dengankan nol masih terdapat 2 fungsi yaitu Y dan w maka persamaan tersebut berbentuk fungsi implisit. Sehingga diperlukan suatu metode numerik untuk dapat menyelesaikannya, maka menurut **DuMouchel, 1999** digunakan metode Newton-Raphson untuk meminimumkan (5.13). Berikut merupakan langkah-langkah metode Newton-Raphson **(Anonim)** yang telah dijelaskan pada sub-bab 2.6 :

Langkah I :

Menentukan nilai awal dari Y_{ij} dan w_i yang dapat ditulis dengan

$$u^0 = \begin{pmatrix} Y_{ij}^{\;0} \\ w_i^{\;0} \end{pmatrix} = \begin{pmatrix} Y_{11}^{\;0} \\ Y_{12}^{\;0} \\ \vdots \\ Y_{MrQ}^{\;0} \\ w_1^{\;0} \\ w_2^{\;0} \\ \vdots \\ w_{Mr}^{\;0} \end{pmatrix} \qquad (5.20)$$

Langkah II :

Menentukan fungsi (5.14) dan (5.17) dalam bentuk matriks, dengan :

$$u^k = \begin{pmatrix} Y_{ij}^{\ 0} \\ w_i^{\ 0} \end{pmatrix} = \begin{pmatrix} Y_{11}^{\ k} \\ Y_{12}^{\ k} \\ \vdots \\ Y_{MrQ}^{\ k} \\ w_1^{\ k} \\ w_2^{\ k} \\ \vdots \\ w_{Mr}^{\ k} \end{pmatrix} \text{ ; k = 0, 1, 2,…,n , yaitu}$$

$$A(u^k) = \begin{pmatrix} (R_{ij}(u^k) \\ (P_i(u^k) \end{pmatrix} \tag{5.21}$$

dimana $(R_{ij}(u^k)$ adalah fungsi dari (5.14) dan $(P_i(u^k)$ adalah fungsi dari (5.17) untuk $i = 1,\ldots,M_r, j = 1,\ldots,Q$

Langkah III :

Menentukan matriks jacobian dari fungsi (5.14) dan (5.17), yaitu

$$J(u^k) = \begin{pmatrix} \frac{\partial R_{11}(u^k)}{\partial Y_{11}} & \cdots & \frac{\partial R_{11}(u^k)}{\partial Y_{M,Q}} & \frac{\partial R_{11}(u^k)}{\partial w_1} & \cdots & \frac{\partial R_{11}(u^k)}{\partial w_1} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial R_{M,Q}(u^k)}{\partial Y_{11}} & \cdots & \frac{\partial R_{M,Q}(u^k)}{\partial Y_{M,Q}} & \frac{\partial R_{M,Q}(u^k)}{\partial w_1} & \cdots & \frac{\partial R_{M,Q}(u^k)}{\partial w_{M_r}} \\ \frac{\partial P_1(u^k)}{\partial Y_{11}} & \cdots & \frac{\partial P_1(u^k)}{\partial Y_{M,Q}} & \frac{\partial P_1(u^k)}{\partial w_1} & \cdots & \frac{\partial P_1(u^k)}{\partial w_{M_r}} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial P_{M_r}(u^k)}{\partial Y_{11}} & \cdots & \frac{\partial P_{M_r}(u^k)}{\partial Y_{M,Q}} & \frac{\partial P_{M_r}(u^k)}{\partial w_1} & \cdots & \frac{\partial P_{M_r}(u^k)}{\partial w_{M_r}} \end{pmatrix} \tag{5.22}$$

dengan :

$$\frac{\partial R_{11}(u^k)}{\partial Y_{11}} = \sum_{k=1}^{K} 2u_k \left[\left(\frac{p_{k1}}{Y_{11}} \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ Pkj} \right)^2 + \left(\frac{p_{k1}}{Y_{11}^{\ 2}} (p_{k1} - 1) \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ Pkj} \right) \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ Pkj} \right) \right]$$

$$\frac{\partial R_{11}(u^k)}{\partial Y_{M_r Q}} = \sum_{k=1}^{K} 2u_k \left[\left(\frac{p_{k1} p_{kQ}}{Y_{11} Y_{M_r Q}} \left(\sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 \right) + \left(\frac{p_{k1} p_{kQ}}{Y_{11} Y_{M_r Q}} \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{11}(u^k)}{\partial w_1} = \sum_{k=1}^{K} 2u_k \left[w_1 \frac{p_{k1}}{Y_{11}} \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{k1}}{Y_{11}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{11}(u^k)}{\partial w_{M_r}} = \sum_{k=1}^{K} 2u_k \left[w_{M_r} \frac{p_{k1}}{Y_{11}} \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{k1}}{Y_{11}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{M_r Q}(u^k)}{\partial Y_{11}} = \sum_{k=1}^{K} 2u_k \left[\left(\frac{p_{k1}}{Y_{11}} \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \left(1 + \frac{p_{kQ}}{Y_{M_r Q}} \right) + \left(\frac{p_{k1}}{Y_{11}^{\,2}} (p_{k1} - 1) \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{M_r Q}(u^k)}{\partial Y_{M_r Q}} = \sum_{k=1}^{K} 2u_k \left[\left(\frac{p_{k1}}{Y_{11}} \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) + \left(\frac{p_{k1}}{Y_{M_r Q}^{\,2}} (p_{k1} - 1) \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{M_r Q}(u^k)}{\partial w_1} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kQ}}{Y_{M_r Q}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{kQ}}{Y_{M_r Q}^{\,2}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial R_{M_r Q}(u^k)}{\partial w_{M_r}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kQ}}{Y_{M_r Q}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{kQ}}{Y_{M_r Q}^{\,2}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial P_1(u^k)}{\partial Y_{11}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{k1}}{Y_{11}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{k1}}{Y_{11}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial P_1(u^k)}{\partial Y_{M_r Q}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kQ}}{Y_{M_r Q}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{kQ}}{Y_{M_r Q}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial P_1(u^k)}{\partial w_1} = \sum_{k=1}^{K} 2u_k \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2$$

$$\frac{\partial P_1(u^k)}{\partial w_{M_r}} = \sum_{k=1}^{K} 2u_k \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2$$

$$\frac{\partial P_{M_r}(u^k)}{\partial Y_{11}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{k1}}{Y_{11}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right)^2 + \frac{p_{k1}}{Y_{11}} \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\,p_{kj}} \right) \right]$$

$$\frac{\partial P_{M_r}(u^k)}{\partial Y_{M_rQ}} = \sum_{k=1}^{K} 2u_k \left[\frac{p_{kQ}}{Y_{M_rQ}} \sum_{i=1}^{Mr} w_i \left(\prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2 + \frac{p_{kQ}}{Y_{M_rQ}} \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \left(z_k - \sum_{i=1}^{Mr} w_i \prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right) \right]$$

$$\frac{\partial P_{M_r}(u^k)}{\partial w_1} = \sum_{k=1}^{K} 2u_k \left(\prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2$$

$$\frac{\partial P_{M_r}(u^k)}{\partial w_{M_r}} = \sum_{k=1}^{K} 2u_k \left(\prod_{j=1}^{Q} Y_{ij}^{\ p_{kj}} \right)^2$$

Langkah IV :

Mencari nilai koreksi (h^k), yaitu :

$$h^k = -\left(J(u^k)\right)^{-1} A(u^k) = \begin{pmatrix} h_{11}^{\ k} \\ \vdots \\ h_{M_rQ}^{\ k} \\ h_{11}^{\ k} \\ \vdots \\ h_{M_rQ}^{\ k} \end{pmatrix} \tag{5.23}$$

dengan $\left(J(u^k)\right)^{-1}$ adalah invers dari $\left(J(u^k)\right)$ untuk $k = 0,1,2,\ldots,n$. Jika matriks $\left(J(u^k)\right)$ singular, maka digunakan invers Moore-Penrose yang dijelaskan pada (**Anggraeni, R, 2002**).

Langkah V :

Menentukan $u^{k+1} = u^k + h^k$ atau $\begin{pmatrix} Y_{11}^{\ k+1} \\ Y_{12}^{\ k+1} \\ \vdots \\ Y_{MrQ}^{\ k+1} \\ w_1^{\ k} \\ w_2^{\ k+1} \\ \vdots \\ w_{Mr}^{\ k+1} \end{pmatrix} = \begin{pmatrix} Y_{11}^{\ k} \\ Y_{12}^{\ k} \\ \vdots \\ Y_{MrQ}^{\ k} \\ w_1^{\ k} \\ w_2^{\ k} \\ \vdots \\ w_{MrQ}^{\ k} \end{pmatrix} + \begin{pmatrix} h_{11}^{\ k} \\ h_{12}^{\ k} \\ \vdots \\ h_{MrQ}^{\ k} \\ h_1^{\ k} \\ h_2^{\ k} \\ \vdots \\ h_{MrQ}^{\ k} \end{pmatrix}$

Dimana u^{k+1} merupakan nilai estimator yang akan dicari.

Langkah VI :

Melakukan pengulangan dari langkah II sampai langkah V hingga nilai koreksi $h = | u^{k+1} - u^k | < error$, dengan error = 0.01. Sehingga diperoleh nilai estimator parameter $\hat{u} = \begin{pmatrix} \hat{Y}_{11}^{k+1} \\ \hat{Y}_{12}^{k+1} \\ \vdots \\ \hat{Y}_{MrQ}^{k+1} \\ \hat{w}_1^{k} \\ \hat{w}_2^{k+1} \\ \vdots \\ \hat{w}_{Mr}^{k+1} \end{pmatrix}$.

4.3 Kasus Khusus

Untuk memudahkan aplikasi dari metode *squashing* maka pada pembahasan berikutnya akan digunakan data satu variabel saja. Persamaan (5.9) untuk kasus satu variabel X=($X_1,\ldots,X_N$) dengan $p_k = k-1$, dan $x = 0$ menjadi :

$$\sum_{i=1}^{M_r} w_i Y_i^{k-1} = \sum_{i=1}^{N_r} X_i^{k-1}, \quad k = 1,\ldots,K \; ; \; r = 1,2,\ldots,R \tag{5.24}$$

dari persamaan (5.24) dengan kondisi :

a. untuk $k = 1$, diperoleh :

$$\sum_{i=1}^{M_r} w_i Y_i^{1-1} = \sum_{i=1}^{N_r} X_i^{1-1} = \sum_{i=1}^{N_r} 1 = N_r$$

atau

$$\sum_{i=1}^{M_r} w_i = N_r \;, \qquad r = 1,2,\ldots,R$$

b. untuk $k=2$, menghitung nilai pengamatan baru (Y_i), dari persamaan (5.24)

diperoleh $\sum_{i=1}^{M_r} w_i Y_i = \sum_{i=1}^{N_r} X_i$

dimisalkan setiap kelompok diambil satu data squashing ($M_r = 1$ untuk $r = 1,2,\ldots,R$) dan untuk r = 1 diperoleh

$$w_1 Y_1 = \sum_{i=1}^{N_1} X_i$$

karena $\sum_{i=1}^{M_1} w_i = N_1$ dan $M_1 = 1$, maka $\sum_{i=1}^{M_1} w_i = \sum_{i=1}^{1} w_i = w_1 = N_1$ sehingga

$w_1 Y_1 = N_1 Y_1 = \sum_{i=1}^{N_1} X_i$ atau $Y_1 = \dfrac{\sum_{i=1}^{N_1} X_i}{N_1} = \overline{X}_1$ untuk $r = 1$.

dengan cara yang sama untuk $r = 2,3,\ldots,R$ diperoleh $w_r = N_r$ dan

$Y_r = \dfrac{\sum_{i=1}^{N_r} X_i}{N_r} = \overline{X}_r$

sehingga secara umum persamaan (5.24) dapat dituliskan sebagai berikut :

$$w_1 Y_1 + w_2 Y_2 + \ldots + w_R Y_R = X_1 + X_2 + \ldots + X_N \quad (5.25)$$

c. Untuk $k=3$, diperoleh :

$$\sum_{i=1}^{M_r} w_i Y_i^{3-1} = \sum_{i=1}^{N_r} X_i^{3-1}$$

atau

$$\sum_{i=1}^{M_r} w_i Y_i^2 = \sum_{i=1}^{N_r} X_i^2$$

karena $\sum_{i=1}^{M_r} w_i = N_1$ dan $M_r = 1$ untuk data lengkap dicentering, maka

$$w_r = N_r \text{ dan } Y_r = \frac{\sum_{i=1}^{N_r} X_i^2}{N_r} \text{ untuk } r = 1,2,\ldots,R.$$

sehingga secara umum persamaan (5.25) dapat dituliskan sebagai berikut :

$$w_1 Y_1^2 + w_2 Y_2^2 + \ldots + w_R Y_R^2 = X_1^2 + X_2^2 + \ldots + X_N^2 \tag{5.26}$$

4.4 Algoritma Program

Algoritma dibuat berdasarkan teori-teori yang telah dibahas dari sub-sub bab sebelumnya. Pada bagian ini akan dibuat beberapa algoritma yang diperlukan dalam pembuatan program :

a. Algoritma untuk membangkitkan data yang berdistribusi Pareto tergeneralisir

1. Membangkitkan data $U_1, U_2, \ldots, U_n$ yang berdistribusi Uniform.
2. Mentransformasi data bangkitan ke dalam variabel acak distribusi Pareto tergeneralisir $x_i = \frac{\beta}{\alpha}\left(\left(1-U_i\right)^{-\alpha} - 1\right)$, dengan x_i adalah variabel acak dan i = 1, 2, …,n.

b. Algoritma metode squashing untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok sama.

1. Memasukkan data induk bangkitan yang berdistribusi Pareto tergeneralisir.
2. Mengurutkan data dari yang terkecil ke terbesar.
3. Mengelompokkan data menjadi beberapa kelompok dengan menggunakan metode kuantil.

4. Menentukan pembobot dari masing-masing kelompok, dimana pembobotnya merupakan jumlah data pada kelompok tersebut.
5. Menentukan *pseudo data point* dari masing-masing kelompok, dengan cara mencari mean dari masing-masing kelompok.
6. Men*squashing* dengan mengalikan antara *pseudo data point* dan pembobot dari masing-masing kelompok.
7. Menggabungkan hasil squashing dari masing-masing kelompok menjadi sub sampel.
8. Mencocokkan jumlahan data *squashing* dengan data induknya.

c. Algoritma metode squashing untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok tidak sama

1. Memasukkan data induk bangkitan yang berdistribusi Pareto tergeneralisir.
2. Menentukan jumlah data yang akan diambil yaitu sebanyak n.
3. Mengurutkan data dari yang terkecil ke terbesar.
4. Menghitung nilai kuantil 0 sampai $\frac{1}{N}$.
5. Mengelompokkan data induk menjadi beberapa kelompok.
6. Menetapkan bahwa kelompok 1 adalah baris pertama, kelompok 2 adalah baris kedua, dan seterusnya.
7. Menentukan pembobot dari masing-masing kelompok, dimana pembobotnya merupakan jumlah data pada kelompok tersebut.
8. Menentukan *pseudo data point* dari masing-masing kelompok, dengan cara mencari mean dari masing-masing kelompok.

9. Men*squashing* dengan mengalikan antara *pseudo data point* dengan pembobot dari masing-masing kelompok.
10. Menggabungkan hasil *squashing* dari masing-masing kelompok menjadi sub sampel.
11. Mencocokkan jumlahan data *squashing* dengan data induknya.

d. Algoritma metode squashing untuk momen ke II (varians) dengan jumlah anggota masing-masing kelompok sama.

1. Memasukkan data induk bangkitan yang berdistribusi Pareto tergeneralisir.
2. Mengurutkan data dari yang terkecil ke terbesar.
3. Mengelompokkan data menjadi beberapa kelompok dengan menggunakan metode kuantil.

1. Menentukan pembobot dari masing-masing kelompok, dimana pembobotnya merupakan jumlah data pada kelompok tersebut.
2. Menentukan *pseudo data point* dari masing-masing kelompok, dengan cara mencari varians dari data yang sudah dicentering untuk masing-masing kelompok.
3. Men*squashing* dengan mengalikan antara *pseudo data point* dan pembobot dari masing-masing kelompok.
4. Menggabungkan hasil *squashing* dari masing-masing kelompok menjadi sub sampel.
5. Mencocokkan jumlahan data *squashing* dengan data lengkapnya.

e. Algoritma metode squashing untuk momen ke II (varians) dengan jumlah anggota masing-masing kelompok tidak sama.

1. Memasukkan data induk bangkitan yang berdistribusi Pareto tergeneralisir.
2. Menentukan jumlah data yang akan diambil yaitu sebanyak n.
3. Mengurutkan data dari yang terkecil ke terbesar.
4. Menghitung nilai kuantil 0 sampai $\frac{1}{N}$.
5. Mengelompokkan data induk menjadi beberapa kelompok.
6. Menetapkan bahwa kelompok 1 adalah baris pertama, kelompok 2 adalah baris kedua, dan seterusnya.
7. Menentukan pembobot dari masing-masing kelompok, dimana pembobotnya merupakan jumlah data pada kelompok tersebut.
8. Menentukan *pseudo data point* dari masing-masing kelompok, dengan cara mencari varians dari data yang sudah dicentering untuk masing-masing kelompok.
9. Men*squashing* dengan mengalikan antara *pseudo data point* dengan pembobot dari masing-masing kelompok.
10. Menggabungkan hasil *squashing* dari masing-masing kelompok menjadi sub sampel.
11. Mencocokkan jumlahan data *squashing* dengan data lengkapnya.

f. Algoritma menentukan mean dan varian dari metode squashing untuk momen ke I (mean) dan momen ke II (varians).

1. Memasukkan data lengkap bangkitan yang berdistribusi Pareto tergeneralisir.
2. Mengurutkan data dari yang terkecil ke terbesar.
2. Mengelompokkan data menjadi beberapa kelompok dengan menggunakan metode kuantil.
3. Menentukan pembobot dari masing-masing kelompok, dimana pembobotnya merupakan jumlah data pada kelompok tersebut.
4. Menentukan *pseudo data point* dari masing-masing kelompok, dengan cara mencari mean dari masing-masing kelompok.
5. Men*squashing* dengan mengalikan antara *pseudo data point* dan pembobot dari masing-masing kelompok.
6. Menentukan nilai mean dan varians dari *squashing* tersebut.
7. Membandingkan mean dan varians data *squashing* dengan data induknya.

5.6 Implementasi Algoritma ke Program

Bahasa pemrograman yang digunakan untuk menerapkan algoritma-algoritma diatas adalah paket program S-Plus, karena S-Plus adalah suatu paket pemrograman berbasis statistik yang didalamnya cukup banyak fungsi standar sehingga memudahkan dalam mengerjakan program.

Berikut ini adalah garis besar program yang digunakan untuk mendukung algoritma yang dibuat, secara rincinya dapat dilihat pada Lampiran 1.

Adapun programnya adalah sebagai berikut :

1. Program **generate** untuk membangkitkan data berdistribusi pareto tergeneralisir dengan asumsi nilai parameternya diketahui.
2. Program **squashing1** merupakan program metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok sama.
3. Program **squashing2** merupakan program metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok tidak sama.
4. Program **squashing3** merupakan program metode *squashing* untuk momen ke II (varians) dengan jumlah anggota masing-masing kelompok sama.
5. Program **squashing4** merupakan program metode *squashing* untuk momen ke II (varians) dengan jumlah anggota masing-masing kelompok tidak sama.
6. Program **squashing5** untuk menentukan nilai mean dan varians metode *squashing* dengan momen ke I (mean).
7. Program **squashing6** untuk menentukan nilai mean dan varians metode *squashing* dengan momen ke II (varians).

5.7 Data

Data yang dipakai untuk mengimplementasikan program yang telah dibuat adalah data bangkitan yang diperoleh dengan cara membangkitkan bilangan acak berdistribusi U(0,1) kemudian ditransformasi ke dalam distribusi Pareto tergeneralisir. Digunakannya data bangkitan ini karena penulis kesulitan untuk mendapatkan data sekunder yang sesuai, mengingat data yang digunakan merupakan data yang berukuran sangat besar. Untuk membangkitkan data berdistribusi Pareto tergeneralisir akan digunakan program **generate**(α,β,n) ; α,β adalah parameter dari distribusi Pareto tergeneralisir dan n merupakan jumlah data (lihat lampiran 1). Dalam pembahasan ini dipilih data berdistribusi Pareto tergeneralisir yang diasumsikan parameter α =0.5, β =0.25 dan jumlah datanya (n) 100.000.

5.8 Analisis Data

Berikut ini merupakan analisa datanya, antara lain adalah sebagai berikut :

1. Membangkitkan data lengkap yang berdistribusi Pareto tergeneralisir menggunakan program **generate** dengan jumlah datanya sebesar 100.000 data *generate*.
2. Mencocokkan momen sub sampel yang terboboti dari data *squashing* sama dengan momen sampel yang tidak terboboti dari data induk.

   Metode *squashing* yang dikelompokkan menjadi dua macam yaitu :

   a. Dikelompokkan menjadi beberapa kelompok dengan jumlah anggota masing-masing kelompok sama. Dimana pada pengelompokkan dengan jumlah yang sama digunakan program **squashing1** dan program **squashing3**.

   b. Dikelompokkan menjadi beberapa kelompok dengan jumlah anggota masing-masing kelompok tidak sama. Dimana pengelompokkan dengan jumlah yang tidak sama digunakan program **squashing2** program **squashing4**.

   Pada program terdapat sq1 yang menunjukkan momen sub sampel yang terboboti dari data *squashing*, sedangkan sq2 merupakan momen sampel yang tidak terboboti dari data induk. Hasil pengolahan data dengan menggunakan program-program diatas dapat dilihat secara lengkap pada lampiran 2. Dari hasil pengolahan tersebut dapat disimpulkan bahwa adanya persamaan antara jumlahan data hasil *squashing* dengan data induknya, baik untuk *squashing* dengan anggota kelompok yang sama dan anggota kelompok yang tidak sama untuk momen ke I dan II.

3. Menentukan nilai mean dan varians dari metode *squashing* untuk momen ke I (mean) digunakan program **squashing5** dan untuk momen ke II (varians) digunakan program **squashing6** (lihat lampiran 1). Pada program terdapat X1 dan V1 yang menunjukkan mean dan varians dari data induknya, sedangkan X2 dan V2 menunjukkan mean dan varians dari metode *squashing*. Hasil dari analisa data dapat dilihat pada tabel 5.1.

Tabel 5.1 Nilai Mean dan Varians Squashing dengan α=0.5, β=0.25 dan jumlah datanya (n) 100.000.

Data	Mean	Varian
Induk	0.4968673	2.818693
Squashing dengan momen I (100 Kelompok)	496.8673	1080150
Squashing dengan momen I (10000 Kelompok)	4.968673	210.4506
Squashing dengan momen I (50000 Kelompok)	0.9937347	10.03842
Squashing dengan momen II (100 Kelompok)	2818.664	612408185
Squashing dengan momen II (10000 Kelompok)	28.18664	2722005
Squashing dengan momen II (50000 Kelompok)	5.637329	339568.3

Dari tabel 5.1 dapat dijelaskan bahwa ketika semakin banyak pengelompokkannya, maka metode *squashing* dengan momen ke I menghasilkan mean dan varians yang mendekati data induknya. Sedangkan untuk metode *squashing* dengan momen ke II menghasilkan mean dan varians yang tidak mendekati data induknya.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan pembahasan dan hasil penerapan data, dapat ditarik kesimpulan bahwa :

1. Mereduksi data dengan menggunakan metode squashing pada data berukuran besar yang berdistribusi Pareto tegeneralisir diperoleh dengan cara sebagai berikut :
   a. Pengelompokkan data ke dalam beberapa daerah atau partisi.
   b. Menentukan pembobot dan *pseudo point* pada setiap partisi.
   c. Hasil reduksi data dengan menggunakan metode *squashing* diindikasikan dengan adanya persamaan antara jumlahan data hasil *squashing* dan data induknya.
2. Penerapan metode *squashing* terhadap data dengan satu variabel, untuk momen I nilai *pseudo point* sama dengan rata-rata per kelompok dan untuk momen II sama dengan variansinya. Sedangkan nilai pembobotnya untuk kedua momen nilainya sama yaitu jumlah anggota per kelompok. Dan aplikasi pada data dengan ukuran 100.000, untuk beberapa pengelompokkan semakin besar jumlah pengelompokkannya maka mean dan varians dari metode *squashing* mendekati sampel induknya.

6.2 Saran

Adapun saran yang ingin disampaikan pada penulisan ini adalah :

1. Penerapan metode *squashing* masih menggunakan data *generate*, diharapkan dapat dicoba pada data sekunder.
2. Program-program yang terdapat pada penelitian ini dapat juga diterapkan pada data yang tidak berdistribusi Pareto tergeneralisir.
3. Untuk penelitian lebih lanjut, pembahasan ini dapat dikembangkan untuk metode *squashing* dengan variabel lebih dari satu.

DAFTAR PUSTAKA

Anggraeni, R. 2002. *Invers Moore-Penrose*, Skripsi, Matematika FMIPA Universitas Airlangga. Surabaya.

Anonim, *Solving A Systems of Nonlinear Equations : Course Notes for 136.212 Introductory Numerical Methods for engineers.* (http://130.179.64.208/intermath/136212/presentcourse/corrections/multiVariableNewtonRaphson.htm).

DuMouchel, W., Chris V.,Theodore J., Corinna C., and Deryl P., 1999, *Squashing Flat Files Flatter*, http://citeseer.nj.nec.com/dumouchel99squashing.html., tgl. Akses 19 Februari 2004.

Dimakos, X.K., 2000, *Data Squashing for Tail Inference in the Generalized Pareto Distribution,* http://www.nr.no/~xeni/phd.html., tgl akses 16 Februari 2004.

Everitt, S. Brian, 1994, *A Handbook of Statistical Analyses Using S-Plus*, Chapman & Hall, London.

Hogg, R.V., Craig, A.T., 1995, *Introduction to Mathematical Statistics*, Fifth Edition, Prentice Hall, inc., New Jersey.

Kartiko, Sri Haryatmi, 1986, *Buku Materi Pokok Pengantar Teori Probabilitas*, Modul : 6-9, Universitas Terbuka , Karunika, Jakarta.

Ross, S.M., 1997, *Introduction to Probability Models*, Sixth Edition, Academic Press, United States of America.

Trehin, P., ---, *Pareto Distribution* , http://perso.wanadoo.fr/gilles.trehin.urville/pareto_distribution.html, tgl. Akses 5 Februari 2004.

Walpole, R.E. and Myers, R.H., 1986, *Ilmu Peluang dan Statistika untuk Insinyur dan Ilmuwan*, Edisi Kedua, Penerjemah : Drs. R. K. Sembiring, ITB, Bandung.

Walpole, R.E., 1995, *Pengantar Statistika*, Edisi Ketiga, Penerjemah : Ir. Bambang Sumantri, PT Gramedia Pustaka Utama, Jakarta.

Lampiran 1. Program

1. Program generate data berdistribusi Pareto tergeneralisir.

```
function(a, b,n)
{
   U <- runif(n)
   P <- (b/a) * (((1 - U)^( - a)) - 1)
   return(P)
}
```

2. Program metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok sama.

```
function(x,n)
{
   x<-as.vector(x)
   x<-sort(x)
   bobot<-rep(0,n)
   squashing<-rep(0,n)
   y<-rep(0,n)
   for(i in 1:n)
   {
      if(i==1)
      {
         q<-x[x>=quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
         cat("kelompok ke-",i,":\n")
         print(q)
         W<-length(q)
         cat("pembobot kelompok ke-",i,":\n")
         print(W)
         y[i]<-sum(q)/W
         cat("var kelompok ke-",i,":\n")
         print(y[i])

      }
      else
      {
         q<-x[x>quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
         cat("kelompok ke-",i,":\n")
         print(q)
         W<-length(q)
         cat("pembobot kelompok ke-",i,":\n")
         print(W)
         y[i]<-sum(q)/W
         cat("var kelompok ke-",i,":\n")
         print(y[i])
      }
      bobot[i]<-W
      sq<-bobot[i]*y[i]
```

```
      cat("squasing kelompok ke-",i,":\n")
      print(sq)
      squashing[i]<-sq

   }
   sq1<-sum(squashing)
   sq2<-sum(x)
   return(bobot,y,squashing,sq1,sq2)
}
```

3. Program metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok tidak sama.

```
function(x,n)
{
   x<-as.vector(x)
   N<-length(x)
   x<-sort(x)
   Q<-quantile(x,probs=seq(0,1,1/N))
   w<-sample(Q,n-1)
   w<-sort(w)
   bobot<-rep(0,n)
   squashing<-rep(0,n)
   y<-rep(0,n)
   for(i in 1:n)
   {
      if(i==1)
      {
         q<-x[x<=w[i]]
      }
      else if(i==n)
      {
         q<-x[x>w[i-1]]
      }

      else
      {
         q<-x[x>w[i-1]&x<=w[i]]
      }
      cat("kel ke-",i,":\n")
      print(q)
      W<-length(q)
      cat("pembobot kelompok ke-",i,":\n")
      print(W)
      y[i]<-sum(q)/W
      cat("var kelompok ke-",i,":\n")
      print(y[i])
      bobot[i]<-W
      sq<-bobot[i]*y[i]
      cat("squasing kelompok ke-",i,":\n")
      print(sq)
      squashing[i]<-sq
   }
```

```
    sq1<-sum(squashing)
    sq2<-sum(x)
    return(bobot,y,squashing,sq1,sq2)
}
```

4. Program metode *squashing* untuk momen ke II (varian) dengan jumlah anggota masing-masing kelompok sama.

```
function(x,n)
{
    x<-as.vector(x)
    sortx<-sort(x)
    x<-(sortx-mean(sortx))^2
    bobot<-rep(0,n)
    squashing<-rep(0,n)
    y<-rep(0,n)
    for(i in 1:n)
    {
        if(i==1)
        {
            q<-x[x>=quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
            cat("kelompok ke-",i,":\n")
            print(q)
            W<-length(q)
                cat("pembobot kelompok ke-",i,":\n")
            print(W)
            y[i]<-sum(q)/W
            cat("var kelompok ke-",i,":\n")
            print(y[i])

        }
        else
        {
            q<-x[x>quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
            cat("kelompok ke-",i,":\n")
            print(q)
            W<-length(q)
            cat("pembobot kelompok ke-",i,":\n")
            print(W)
            y[i]<-sum(q)/W
            cat("var kelompok ke-",i,":\n")
            print(y[i])
        }
        bobot[i]<-W
        sq<-bobot[i]*y[i]
        cat("squasing kelompok ke-",i,":\n")
        print(sq)
        squashing[i]<-sq

    }
```

```
   sq1<-sum(squashing)
   sq2<-sum(x)
   return(bobot,y,squashing,sq1,sq2)
}
```

5. Program metode *squashing* untuk momen ke II (varian) dengan jumlah anggota masing-masing kelompok tidak sama.

```
function(x,n)
{
   x<-as.vector(x)
   N<-length(x)
   sortx<-sort(x)
   x<-(sortx-mean(sortx))^2
   Q<-quantile(x,probs=seq(0,1,1/N))
   w<-sample(Q,n-1)
   w<-sort(w)
   bobot<-rep(0,n)
   squashing<-rep(0,n)
   y<-rep(0,n)
   for(i in 1:n)
   {
      if(i==1)
      {
         q<-x[x<=w[i]]
      }
      else if(i==n)
      {
         q<-x[x>w[i-1]]
      }
      else
      {
         q<-x[x>w[i-1]&x<=w[i]]
      }
      cat("kel ke-",i,":\n")
      print(q)
      W<-length(q)
      cat("pembobot kelompok ke-",i,":\n")
      print(W)
      y[i]<-sum(q)/W
      cat("var kelompok ke-",i,":\n")
      print(y[i])
      bobot[i]<-W
      sq<-bobot[i]*y[i]
      cat("squasing kelompok ke-",i,":\n")
      print(sq)
      squashing[i]<-sq

   }
   sq1<-sum(squashing)
   sq2<-sum(x)
   return(bobot,y,squashing,sq1,sq2)
}
```

6. Program untuk menentukan nilai mean dan varians metode *squashing* dengan momen ke I (mean).

```
function(x,n)
{
   x<-as.vector(x)
   x<-sort(x)
   bobot<-rep(0,n)
   squashing<-rep(0,n)
   y<-rep(0,n)
   for(i in 1:n)
   {
      if(i==1)
      {
         q<-x[x>=quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
         cat("kelompok ke-",i,":\n")
         print(q)
         W<-length(q)
         cat("pembobot kelompok ke-",i,":\n")
         print(W)
         y[i]<-sum(q)/W
         cat("var kelompok ke-",i,":\n")
         print(y[i])


      }
      else
      {
         q<-x[x>quantile(x,prob=((i-1)/n))&x<=quantile(x,prob=(i/n))]
         cat("kelompok ke-",i,":\n")
         print(q)
         W<-length(q)
         cat("pembobot kelompok ke-",i,":\n")
         print(W)
         y[i]<-sum(q)/W
         cat("var kelompok ke-",i,":\n")
         print(y[i])


      }
      bobot[i]<-W
      sq<-bobot[i]*y[i]
      cat("squasing kelompok ke-",i,":\n")
      print(sq)
      squashing[i]<-sq

   }
   X1<-mean(x)
   V1<-var(x)
   X2<-mean(squashing)
   V2<-var(squashing)
   return(bobot,y,squashing,X1,V1,X2,V2)

}
```

7. Program untuk menentukan nilai mean dan varians metode *squashing* dengan momen ke I (mean).

```
function(x,n)
{
    x<-as.vector(x)
    sortx<-sort(x)
    x<-(sortx-mean(sortx))^2
    bobot<-rep(0,n)
    squashing<-rep(0,n)
    y<-rep(0,n)
    for(i in 1:n)
    {
        if(i==1)
        {
            q<-x[x>=quantile(x,prob=((i-
1)/n))&x<=quantile(x,prob=(i/n))]
            cat("kelompok ke-",i,":\n")
            print(q)
            W<-length(q)
            cat("pembobot kelompok ke-",i,":\n")
            print(W)
            y[i]<-sum(q)/W
            cat("var kelompok ke-",i,":\n")
            print(y[i])


        }
        else
        {
            q<-x[x>quantile(x,prob=((i-1)/n))&x<=quantile(x,prob=(i/n))]
            cat("kelompok ke-",i,":\n")
            print(q)
            W<-length(q)
            cat("pembobot kelompok ke-",i,":\n")
            y[i]<-sum(q)/W
            cat("var kelompok ke-",i,":\n")
            print(y[i])


        }
        bobot[i]<-W
        sq<-bobot[i]*y[i]
        cat("squasing kelompok ke-",i,":\n")
        print(sq)
        squashing[i]<-sq

    }
    X1<-mean(x)
    V1<-var(x)
    X2<-mean(squashing)
    V2<-var(squashing)
    return(bobot,y,squashing,X1,V1,X2,V2)
}
```

Lampiran 2. Output Program

1. Metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok sama.

```
$bobot:
  [1] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [15] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [29] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [43] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [57] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [71] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [85] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
 [99] 1000 1000

$y:
  [1] 0.001321143 0.003979116 0.006576717 0.009174785 0.011867199 0.014568282
  [7] 0.017480703 0.020342033 0.023140808 0.026029692 0.028893396 0.031688859
 [13] 0.034608143 0.037736426 0.040858137 0.044138804 0.047339523 0.050686786
 [19] 0.054287613 0.057732987 0.061253908 0.064938541 0.068597051 0.072309626
 [25] 0.076143303 0.079940864 0.083778510 0.087766033 0.091871847 0.096026743
 [31] 0.100280514 0.104630981 0.109081095 0.113820925 0.118439264 0.123092216
 [37] 0.127882990 0.133005013 0.138447040 0.143659316 0.149027137 0.154465322
 [43] 0.159837293 0.165606747 0.171878446 0.178328069 0.184662107 0.191716734
 [49] 0.198674449 0.205581116 0.212973152 0.220308802 0.228204209 0.235859286
 [55] 0.243992695 0.252641459 0.261099664 0.269830065 0.279182982 0.288764698
 [61] 0.298692441 0.308876770 0.319066647 0.330020570 0.341425656 0.353119359
 [67] 0.365950726 0.379181267 0.393007868 0.407136742 0.422368573 0.438240672
 [73] 0.455573014 0.473436383 0.492882298 0.512788568 0.533240442 0.555339170
 [79] 0.580133460 0.606220507 0.634010094 0.662892360 0.694249104 0.729358550
 [85] 0.768399139 0.810289940 0.858621255 0.911056515 0.972677704 1.042106961
 [91] 1.122224366 1.217299383 1.325369565 1.462678494 1.642909203 1.877434053
 [97] 2.197312974 2.681036290 3.668113712 9.031941124

$squashing:
  [1]    1.321143    3.979116    6.576717    9.174785   11.867199   14.568282
  [7]   17.480703   20.342033   23.140808   26.029692   28.893396   31.688859
 [13]   34.608143   37.736426   40.858137   44.138804   47.339523   50.686786
 [19]   54.287613   57.732987   61.253908   64.938541   68.597051   72.309626
 [25]   76.143303   79.940864   83.778510   87.766033   91.871847   96.026743
 [31]  100.280514  104.630981  109.081095  113.820925  118.439264  123.092216
 [37]  127.882990  133.005013  138.447040  143.659316  149.027137  154.465322
 [43]  159.837293  165.606747  171.878446  178.328069  184.662107  191.716734
 [49]  198.674449  205.581116  212.973152  220.308802  228.204209  235.859286
 [55]  243.992695  252.641459  261.099664  269.830065  279.182982  288.764698
 [61]  298.692441  308.876770  319.066647  330.020570  341.425656  353.119359
 [67]  365.950726  379.181267  393.007868  407.136742  422.368573  438.240672
 [73]  455.573014  473.436383  492.882298  512.788568  533.240442  555.339170
 [79]  580.133460  606.220507  634.010094  662.892360  694.249104  729.358550
 [85]  768.399139  810.289940  858.621255  911.056515  972.677704 1042.106961
 [91] 1122.224366 1217.299383 1325.369565 1462.678494 1642.909203 1877.434053
 [97] 2197.312974 2681.036290 3668.113712 9031.941124
```

```
$sq1:
[1] 49686.73

$sq2:
[1] 49686.73
```

2. Metode *squashing* untuk momen ke I (mean) dengan jumlah anggota masing-masing kelompok tidak sama.

```
$bobot:
 [1] 1095  102 1065 1919  811  524 3363 2306 1381  458 1152  135  910  178
[15]  435 3171 1422  238 1169  594  529 1343 1924 1619   21  845 1524  418
[29]   67  179  195 1009 1668  234  421 1160 1713 1125  991 3640  973 1704
[43]   78 1115  262 1004  953  354 1210  154  476  823   58 1520  365 1461
[57]  764 3750  440 1441 3000 1789  315 1073  488  695  484 1030  268 1083
[71] 1076  812  382  754  484 3388  781  431  320 1021  184  197   28   67
[85]  519 2426  501  735 2184 3871   97  832 1470  343  870   97  136 3004
[99]  390  417

$y:
 [1] 0.001451730 0.003059071 0.004570404 0.008455895  0.012105103
 [6] 0.013878519 0.019451128 0.027546163 0.032777114  0.035488047
[11] 0.038052745 0.040043528 0.041701977 0.043496219  0.044527112
[16] 0.050506200 0.058497186 0.061406257 0.064001047  0.067231599
[21] 0.069313624 0.072802707 0.079038531 0.085899608  0.089209042
[26] 0.090988060 0.095901149 0.099982091 0.101093693  0.101735424
[31] 0.102514503 0.105030141 0.111157102 0.115575607  0.117112430
[36] 0.120784382 0.127616887 0.134934920 0.140644213  0.153003092
[41] 0.165843861 0.174321602 0.180085396 0.183786200  0.188580089
[46] 0.193025663 0.199810701 0.204332563 0.209972286  0.215101721
[51] 0.217350824 0.222164365 0.225655736 0.231862836  0.239242441
[56] 0.246868348 0.256421929 0.276549863 0.296613510  0.306206011
[61] 0.329706734 0.357653103 0.371181411 0.380502280  0.391381033
[66] 0.399464319 0.408053656 0.419513168 0.429694781  0.440635373
[71] 0.459459408 0.476507485 0.488121091 0.499439958  0.511744398
[76] 0.553790467 0.605341066 0.621915666 0.632815336  0.651679728
[81] 0.669770946 0.675614437 0.679098203 0.680780384  0.689974927
[86] 0.742930187 0.801607517 0.829484129 0.905864628  1.122636523
[91] 1.312495379 1.368974498 1.550142678 1.733844578  1.881446193
[96] 2.026844371 2.062655925 3.022985880 5.839300592 13.984887670

$squashing:
 [1]   1.5896448   0.3120253   4.8674801  16.2268626   9.8172386
 [6]   7.2723438  65.4141438  63.5214515  45.2651939  16.2535253
[11]  43.8367620   5.4058763  37.9487990   7.7423270  19.3692937
[16] 160.1551598  83.1829989  14.6146893  74.8172245  39.9355700
[21]  36.6669071  97.7740354 152.0701328 139.0714652   1.8733899
[26]  76.8849104 146.1533509  41.7925142   6.7732774  18.2106409
[31]  19.9903280 105.9754122 185.4100455  27.0446920  49.3043329
[36] 140.1098825 218.6077271 151.8017845 139.3784149 556.9312544
[41] 161.3660768 297.0440104  14.0466609 204.9216135  49.4079832
```

```
[46]  193.7977654  190.4195977   72.3337274  254.0664659   33.1256650
[51]  103.4589923  182.8412723   13.0880327  352.4315102   87.3234909
[56]  360.6746566  195.9063541 1037.0619870  130.5099442  441.2428621
[61]  989.1202017  639.8414011  116.9221443  408.2789465  190.9939441
[66]  277.6277017  197.4979696  432.0985633  115.1582014  477.2081088
[71]  494.3783229  386.9240782  186.4622567  376.5777282  247.6842886
[76] 1876.2421020  472.7713728  268.0456519  202.5009075  665.3650022
[81]  123.2378541  133.0960441   19.0147497   45.6122857  358.0969870
[86] 1802.3486333  401.6053659  609.6708346 1978.4083486 4345.7259816
[91]  127.3120517 1138.9867825 2278.7097363  594.7086901 1636.8581877
[96]  196.6039040  280.5212058 9081.0495849 2277.3272308 5831.6981585

$sq1:
[1] 49686.73

$sq2:
[1] 49686.73
```

3. Metode *squashing* untuk momen ke II (varian) dengan jumlah anggota masing-masing kelompok sama.

```
$bobot:
 [1] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[16] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[31] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[46] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[61] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[76] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[91] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000

$y:
  [1] 3.291033e-005 2.309604e-004 6.177108e-004 1.178087e-003 1.933110e-003
  [6] 2.884986e-003 4.027735e-003 5.342246e-003 6.813446e-003 8.494235e-003
 [11] 1.022402e-002 1.218030e-002 1.428016e-002 1.648848e-002 1.887866e-002
 [16] 2.133300e-002 2.380456e-002 2.633757e-002 2.920887e-002 3.193537e-002
 [21] 3.474128e-002 3.769758e-002 4.072219e-002 4.383691e-002 4.704842e-002
 [26] 5.038687e-002 5.370269e-002 5.692348e-002 6.027649e-002 6.377779e-002
 [31] 6.727306e-002 7.064415e-002 7.425071e-002 7.781497e-002 8.135971e-002
 [36] 8.495179e-002 8.838372e-002 9.196975e-002 9.571092e-002 9.931568e-002
 [41] 1.028044e-001 1.064832e-001 1.100910e-001 1.135255e-001 1.168449e-001
 [46] 1.201965e-001 1.236384e-001 1.269583e-001 1.304773e-001 1.340242e-001
 [51] 1.373627e-001 1.405632e-001 1.437877e-001 1.469609e-001 1.503085e-001
 [56] 1.535265e-001 1.566815e-001 1.598808e-001 1.629411e-001 1.660932e-001
 [61] 1.691724e-001 1.721699e-001 1.751169e-001 1.781579e-001 1.811703e-001
 [66] 1.841743e-001 1.871301e-001 1.901245e-001 1.930373e-001 1.959043e-001
 [71] 1.989130e-001 2.017747e-001 2.044937e-001 2.073571e-001 2.100870e-001
 [76] 2.128877e-001 2.154905e-001 2.180160e-001 2.205858e-001 2.231896e-001
 [81] 2.257572e-001 2.283526e-001 2.309884e-001 2.336309e-001 2.361463e-001
 [86] 2.386514e-001 2.411156e-001 2.435761e-001 2.472869e-001 2.976940e-001
 [91] 3.917412e-001 5.198798e-001 6.875366e-001 9.348830e-001 1.317185e+000
 [96] 1.911900e+000 2.902729e+000 4.803484e+000 1.023144e+001 2.475433e+002
```

```
$squashing:
 [1] 3.291033e-002  2.309604e-001  6.177108e-001  1.178087e+000  1.933110e+000
 [6] 2.884986e+000  4.027735e+000  5.342246e+000  6.813446e+000  8.494235e+000
[11] 1.022402e+001  1.218030e+001  1.428016e+001  1.648848e+001  1.887866e+001
[16] 2.133300e+001  2.380456e+001  2.633757e+001  2.920887e+001  3.193537e+001
[21] 3.474128e+001  3.769758e+001  4.072219e+001  4.383691e+001  4.704842e+001
[26] 5.038687e+001  5.370269e+001  5.692348e+001  6.027649e+001  6.377779e+001
[31] 6.727306e+001  7.064415e+001  7.425071e+001  7.781497e+001  8.135971e+001
[36] 8.495179e+001  8.838372e+001  9.196975e+001  9.571092e+001  9.931568e+001
[41] 1.028044e+002  1.064832e+002  1.100910e+002  1.135255e+002  1.168449e+002
[46] 1.201965e+002  1.236384e+002  1.269583e+002  1.304773e+002  1.340242e+002
[51] 1.373627e+002  1.405632e+002  1.437877e+002  1.469609e+002  1.503085e+002
[56] 1.535265e+002  1.566815e+002  1.598808e+002  1.629411e+002  1.660932e+002
[61] 1.691724e+002  1.721699e+002  1.751169e+002  1.781579e+002  1.811703e+002
[66] 1.841743e+002  1.871301e+002  1.901245e+002  1.930373e+002  1.959043e+002
[71] 1.989130e+002  2.017747e+002  2.044937e+002  2.073571e+002  2.100870e+002
[76] 2.128877e+002  2.154905e+002  2.180160e+002  2.205858e+002  2.231896e+002
[81] 2.257572e+002  2.283526e+002  2.309884e+002  2.336309e+002  2.361463e+002
[86] 2.386514e+002  2.411156e+002  2.435761e+002  2.472869e+002  2.976940e+002
[91] 3.917412e+002  5.198798e+002  6.875366e+002  9.348830e+002  1.317185e+003
[96] 1.911900e+003  2.902729e+003  4.803484e+003  1.023144e+004  2.475433e+005

$sq1:
[1] 281866.4

$sq2:
[1] 281866.4
```

4. Metode *squashing* untuk momen ke II (varian) dengan jumlah anggota masing-masing kelompok tidak sama.

```
$bobot:
 [1] 1095  102 1065 1919  811  524 3363 2306 1381  458 1152  135  910  178  435
[16] 3171 1422  238 1169  594  529 1343 1924 1619   21  845 1524  418   67  179
[31]  195 1009 1668  234  421 1160 1713 1125  991 3640  973 1704   78 1115  262
[46] 1004  953  354 1210  154  476  823   58 1520  365 1461  764 3750  440 1441
[61] 3000 1789  315 1073  488  695  484 1030  268 1083 1076  812  382  754  484
[76] 3388  781  431  320 1021  184  197   28   67  519 2426  501  735 2184 3871
[91]   97  832 1470  343  870   97  136 3004  390  417

$y:
 [1] 3.958940e-005 1.314795e-004 3.035664e-004 1.024521e-003 2.007096e-003
 [6] 2.631089e-003 4.994705e-003 9.422112e-003 1.296637e-002 1.490279e-002
[11] 1.672732e-002 1.682774e-002 1.952286e-002 2.086490e-002 2.161901e-002
[16] 2.624206e-002 3.253662e-002 3.485158e-002 3.695309e-002 3.957868e-002
[21] 4.132520e-002 4.424119e-002 4.958835e-002 5.542833e-002 5.802679e-002
[26] 5.953612e-002 6.367368e-002 6.705046e-002 6.790693e-002 6.838232e-002
[31] 6.901865e-002 7.096533e-002 7.581620e-002 7.913641e-002 8.030158e-002
```

[56] 1.514014e-001 1.548614e-001 1.620408e-001 1.685498e-001 1.713881e-001
[61] 1.780220e-001 1.852035e-001 1.884156e-001 1.904073e-001 1.926740e-001
[66] 1.944066e-001 1.960586e-001 1.983653e-001 2.002785e-001 2.021516e-001
[71] 2.051283e-001 2.078206e-001 2.093997e-001 2.110135e-001 2.127555e-001
[76] 2.177411e-001 2.231106e-001 2.246633e-001 2.256307e-001 2.273644e-001
[81] 2.289661e-001 2.294422e-001 2.297308e-001 2.298509e-001 2.306102e-001
[86] 2.344795e-001 2.381791e-001 2.396697e-001 2.433011e-001 4.019026e-001
[91] 6.652570e-001 7.615649e-001 1.115282e+000 1.530630e+000 1.921609e+000
[96] 2.340912e+000 2.451868e+000 6.999776e+000 2.884812e+001 5.581396e+002

$squashing:
[1] 4.335039e-002 1.341091e-002 3.232982e-001 1.966056e+000 1.627755e+000
[6] 1.378691e+000 1.679719e+001 2.172739e+001 1.790655e+001 6.825477e+000
[11] 1.926987e+001 2.467452e+000 1.776580e+001 3.713953e+000 9.404268e+000
[16] 8.321358e+001 4.626708e+001 8.294675e+000 4.319816e+001 2.350974e+001
[21] 2.186103e+001 5.941592e+001 9.540798e+001 8.973847e+001 1.218563e+000
[26] 5.030802e+001 9.703869e+001 2.802709e+001 4.549765e+000 1.224044e+001
[31] 1.345864e+001 7.160401e+001 1.264614e+002 1.851792e+001 3.380697e+001
[36] 9.650724e+001 1.510740e+002 1.049617e+002 9.638821e+001 3.840274e+002
[41] 1.105869e+002 2.012597e+002 9.443869e+000 1.373395e+002 3.288157e+001
[46] 1.280978e+002 1.249137e+002 4.721882e+001 1.646256e+002 2.129934e+001
[51] 6.629331e+001 1.163475e+002 8.281505e+000 2.208342e+002 5.416800e+001
[56] 2.211974e+002 1.183141e+002 6.076531e+002 7.416191e+001 2.469702e+002
[61] 5.340660e+002 3.313291e+002 5.935093e+001 2.043071e+002 9.402489e+001
[66] 1.351126e+002 9.489236e+001 2.043163e+002 5.367463e+001 2.189302e+002
[71] 2.207181e+002 1.687503e+002 7.999070e+001 1.591042e+002 1.029737e+002
[76] 7.377068e+002 1.742494e+002 9.682988e+001 7.220181e+001 2.321391e+002
[81] 4.212977e+001 4.520012e+001 6.432462e+000 1.540001e+001 1.196867e+002
[86] 5.688472e+002 1.193277e+002 1.761572e+002 5.313697e+002 1.555765e+003
[91] 6.452993e+001 6.336220e+002 1.639464e+003 5.250062e+002 1.671800e+003
[96] 2.270685e+002 3.334541e+002 2.102733e+004 1.125077e+004 2.327442e+005

$sq1:
[1] 281866.4

$sq2:
[1] 281866.4

5. Nilai mean dan varians metode *squashing* dengan momen ke I (mean) untuk 100 kelompok.

$bobot:
[1] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[15] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[29] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[43] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[57] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[71] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[85] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[99] 1000 1000

```
$y:
 [1] 0.001321143 0.003979116 0.006576717 0.009174785 0.011867199 0.014568282
 [7] 0.017480703 0.020342033 0.023140808 0.026029692 0.028893396 0.031688859
[13] 0.034608143 0.037736426 0.040858137 0.044138804 0.047339523 0.050686786
[19] 0.054287613 0.057732987 0.061253908 0.064938541 0.068597051 0.072309626
[25] 0.076143303 0.079940864 0.083778510 0.087766033 0.091871847 0.096026743
[31] 0.100280514 0.104630981 0.109081095 0.113820925 0.118439264 0.123092216
[37] 0.127882990 0.133005013 0.138447040 0.143659316 0.149027137 0.154465322
[43] 0.159837293 0.165606747 0.171878446 0.178328069 0.184662107 0.191716734
[49] 0.198674449 0.205581116 0.212973152 0.220308802 0.228204209 0.235859286
[55] 0.243992695 0.252641459 0.261099664 0.269830065 0.279182982 0.288764698
[61] 0.298692441 0.308876770 0.319066647 0.330020570 0.341425656 0.353119359
[67] 0.365950726 0.379181267 0.393007868 0.407136742 0.422368573 0.438240672
[73] 0.455573014 0.473436383 0.492882298 0.512788568 0.533240442 0.555339170
[79] 0.580133460 0.606220507 0.634010094 0.662892360 0.694249104 0.729358550
[85] 0.768399139 0.810289940 0.858621255 0.911056515 0.972677704 1.042106961
[91] 1.122224366 1.217299383 1.325369565 1.462678494 1.642909203 1.877434053
[97] 2.197312974 2.681036290 3.668113712 9.031941124

$squashing:
 [1]    1.321143    3.979116    6.576717    9.174785   11.867199   14.568282
 [7]   17.480703   20.342033   23.140808   26.029692   28.893396   31.688859
[13]   34.608143   37.736426   40.858137   44.138804   47.339523   50.686786
[19]   54.287613   57.732987   61.253908   64.938541   68.597051   72.309626
[25]   76.143303   79.940864   83.778510   87.766033   91.871847   96.026743
[31]  100.280514  104.630981  109.081095  113.820925  118.439264  123.092216
[37]  127.882990  133.005013  138.447040  143.659316  149.027137  154.465322
[43]  159.837293  165.606747  171.878446  178.328069  184.662107  191.716734
[49]  198.674449  205.581116  212.973152  220.308802  228.204209  235.859286
[55]  243.992695  252.641459  261.099664  269.830065  279.182982  288.764698
[61]  298.692441  308.876770  319.066647  330.020570  341.425656  353.119359
[67]  365.950726  379.181267  393.007868  407.136742  422.368573  438.240672
[73]  455.573014  473.436383  492.882298  512.788568  533.240442  555.339170
[79]  580.133460  606.220507  634.010094  662.892360  694.249104  729.358550
[85]  768.399139  810.289940  858.621255  911.056515  972.677704 1042.106961
[91] 1122.224366 1217.299383 1325.369565 1462.678494 1642.909203 1877.434053
[97] 2197.312974 2681.036290 3668.113712 9031.941124

$X1:
[1] 0.4968673

$V1:
[1] 2.818693

$X2:
[1] 496.8673

$V2:
[1] 1080150
```

6. Nilai mean dan varians metode *squashing* dengan momen ke II (varian) untuk 100 kelompok.

```
$bobot:
 [1]  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[16] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[31] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[46] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[61] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[76] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
[91] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000

$y:
  [1] 3.291033e-005  2.309604e-004 6.177108e-004 1.178087e-003 1.933110e-003
  [6] 2.884986e-003  4.027735e-003 5.342246e-003 6.813446e-003 8.494235e-003
 [11] 1.022402e-002 1.218030e-002 1.428016e-002 1.648848e-002 1.887866e-002
 [16] 2.133300e-002 2.380456e-002 2.633757e-002 2.920887e-002 3.193537e-002
 [21] 3.474128e-002 3.769758e-002 4.072219e-002 4.383691e-002 4.704842e-002
 [26] 5.038687e-002 5.370269e-002 5.692348e-002 6.027649e-002 6.377779e-002
 [31] 6.727306e-002 7.064415e-002 7.425071e-002 7.781497e-002 8.135971e-002
 [36] 8.495179e-002 8.838372e-002 9.196975e-002 9.571092e-002 9.931568e-002
 [41] 1.028044e-001 1.064832e-001 1.100910e-001 1.135255e-001 1.168449e-001
 [46] 1.201965e-001 1.236384e-001 1.269583e-001 1.304773e-001 1.340242e-001
 [51] 1.373627e-001 1.405632e-001 1.437877e-001 1.469609e-001 1.503085e-001
 [56] 1.535265e-001 1.566815e-001 1.598808e-001 1.629411e-001 1.660932e-001
 [61] 1.691724e-001 1.721699e-001 1.751169e-001 1.781579e-001 1.811703e-001
 [66] 1.841743e-001 1.871301e-001 1.901245e-001 1.930373e-001 1.959043e-001
 [71] 1.989130e-001 2.017747e-001 2.044937e-001 2.073571e-001 2.100870e-001
 [76] 2.128877e-001 2.154905e-001 2.180160e-001 2.205858e-001 2.231896e-001
 [81] 2.257572e-001 2.283526e-001 2.309884e-001 2.336309e-001 2.361463e-001
 [86] 2.386514e-001 2.411156e-001 2.435761e-001 2.472869e-001 2.976940e-001
 [91] 3.917412e-001 5.198798e-001 6.875366e-001 9.348830e-001 1.317185e+000
 [96] 1.911900e+000 2.902729e+000 4.803484e+000 1.023144e+001 2.475433e+002

$squashing:
  [1] 3.291033e-002  2.309604e-001  6.177108e-001  1.178087e+000  1.933110e+000
  [6] 2.884986e+000  4.027735e+000  5.342246e+000  6.813446e+000  8.494235e+000
 [11] 1.022402e+001  1.218030e+001  1.428016e+001  1.648848e+001  1.887866e+001
 [16] 2.133300e+001  2.380456e+001  2.633757e+001  2.920887e+001  3.193537e+001
 [21] 3.474128e+001  3.769758e+001  4.072219e+001  4.383691e+001  4.704842e+001
 [26] 5.038687e+001  5.370269e+001  5.692348e+001  6.027649e+001  6.377779e+001
 [31] 6.727306e+001  7.064415e+001  7.425071e+001  7.781497e+001  8.135971e+001
 [36] 8.495179e+001  8.838372e+001  9.196975e+001  9.571092e+001  9.931568e+001
 [41] 1.028044e+002  1.064832e+002  1.100910e+002  1.135255e+002  1.168449e+002
 [46] 1.201965e+002  1.236384e+002  1.269583e+002  1.304773e+002  1.340242e+002
 [51] 1.373627e+002  1.405632e+002  1.437877e+002  1.469609e+002  1.503085e+002
 [56] 1.535265e+002  1.566815e+002  1.598808e+002  1.629411e+002  1.660932e+002
 [61] 1.691724e+002  1.721699e+002  1.751169e+002  1.781579e+002  1.811703e+002
 [66] 1.841743e+002  1.871301e+002  1.901245e+002  1.930373e+002  1.959043e+002
 [71] 1.989130e+002  2.017747e+002  2.044937e+002  2.073571e+002  2.100870e+002
 [76] 2.128877e+002  2.154905e+002  2.180160e+002  2.205858e+002  2.231896e+002
 [81] 2.257572e+002  2.283526e+002  2.309884e+002  2.336309e+002  2.361463e+002
 [86] 2.386514e+002  2.411156e+002  2.435761e+002  2.472869e+002  2.976940e+002
 [91] 3.917412e+002  5.198798e+002  6.875366e+002  9.348830e+002  1.317185e+003
```

[96] 1.911900e+003 2.902729e+003 4.803484e+003 1.023144e+004 2.475433e+005

$X1:
[1] 2.818664

$V1:
[1] 145256.8

$X2:
[1] 2818.664

$V2:
[1] 612408185