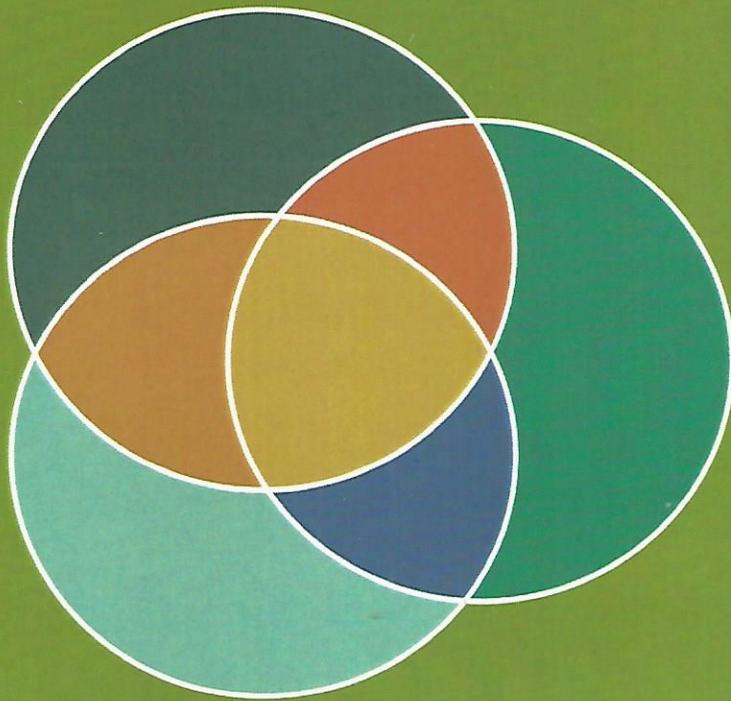


Jurnal
MSA
Matematika dan Statistika serta Aplikasinya



JURUSAN MATEMATIKA
FST- UIN ALAUDDIN MAKASSAR

TUTORIAL SOFTWARE: ANALISIS DATA MENGGUNAKAN PEMROGRAMAN MAKRO MINITAB

Muh. Irwanⁱ

ⁱ Program Studi Matematika FST UINAM, muhirwan@uin-alauddin.ac.id

ABSTRAK, Minitab merupakan salah satu paket program pengolahan data statistik yang sangat baik dan digemari oleh statistisi maupun ahli teknik. Minitab memiliki bahasa pemrograman yang dikenal dengan macro minitab. Pada bagian pembahasan telah dijelaskan operator, perulangan, *control statement* dan tipe data sebagai dasar pemrograman lanjutan dalam minitab. Sehingga minitab menjadi alternatif bahasa pemrograman yang dapat digunakan dalam menjelaskan data dalam kehidupan sehari-hari.

Kata Kunci: Minitab, Macro, Global, Local

1. PENDAHULUAN

Minitab merupakan salah satu paket program pengolahan data statistik yang sangat baik dan digemari oleh statistisi maupun ahli teknik. Minitab memiliki banyak keunggulan dibandingkan dengan paket program lainnya. Kemampuan dan ketangguhan paket program ini meliputi hampir semua alat analisis statistik yaitu: Basic Statistics yang meliputi: Descriptive, 1 sample dan Paired parametric test, Proportion, correlation, Covariance dan Normality, Regression yang meliputi : linear regression, stepwise, best subset, fitted line plot dan logistic regression. Ketangguhan lain adalah ANOVA yang sangat lengkap, Design Of Experiment (DOE) yang dilengkapi dengan metode Taguchi, Control Chart, Quality tools dan Reliability yang meliputi seluruh metode pengendalian kualitas dan reliabilitas[1].

Multivariate Analysis dengan metodenya analisis komponen utama (Principal Component Analysis), Analisis Faktor, Analisis Kluster dan Analisis Deskriminan sungguh sangat berarti bagi penyelesaian masalah analisis multivariat. Disamping itu, analisis time series yang sangat komplis yaitu analisis trend, dekomposisi, moving average, smoothing, winters dan metode ARIMA sangat cukup untuk analisis data berkala runtun waktu. Metode nonparametrik juga sangat lengkap, terdiri dari uji SINGS (sign test),

Wilcoxon Test, Mann Whitney Test, Kruskall Walls, Mood's Median Test, Friedmann, Runs Test, Pairwised Average, Difference dan Slope[1].

Minitab memiliki keunggulan lain berupa kemudahan untuk pengoperasiannya. Sehingga sangat membantu penyelesaian masalah statistik bagi siapa saja, tidak peduli apakah ia sudah mahir komputer atau belum. Cukup dengan sedikit latihan, sudah dapat menyelesaikan permasalahannya. Disamping itu, informasi yang disajikan oleh Minitab sangat banyak dan lengkap. Selain itu, Minitab memiliki Bahasa pemrograman tersendiri yang dikenal dengan Bahasa macro minitab. Untuk memudahkan pemahaman tentang macro minitab, pada akhir artikel ini akan diperlihatkan satu contoh kasus mengenai ukuran penyebaran dan ukuran pemusatan statistik deskriptif.

2. TINJAUAN PUSTAKA

Macro Minitab

Macro Minitab adalah perintah atau rangkaian perintah (command) yang membentuk fungsi tertentu (biasanya lebih khusus) dan tidak disediakan oleh Minitab. Untuk membuat macro minitab sangat beragam caranya namun umumnya dibuat dengan *text editor* seperti notepad karena Minitab tidak menyediakan window tersendiri untuk membuat macro. Macro yang dibuat dapat dijalankan pada window Session yang telah diaktifkan sebelumnya dengan cara *Editor > Enable Command*. Dan tampilan window Session dapat diatur pada *Tools > Options > Session Window*[3].

Minitab pada dasarnya untuk memenuhi kebutuhan analisis data yang kompleks dan perlu algoritma tertentu yang tidak terdapat pada fasilitas Minitab.

Terdapat tiga jenis macro yang dikembangkan oleh Minitab untuk melakukan beragam tugas yang berulang (*repetitive*) dengan mudah dan efektif, yaitu, Global Macro, Execs dan Local Macro. Namun pada tulisan ini hanya akan dijelaskan tentang Local Macro [3].

Local macro

Local Macro memiliki struktur yang lebih kompleks dan lebih rumit namun lebih canggih dan fleksibel daripada global macro. Local macro menggunakan variabel sementara (*temporary*), argumen, dan subcommand (atau bisa disebut elemen) [2][3]. Elemen tersebut berguna untuk meningkatkan kemampuan dari macro sehingga struktur yang digunakan berbeda dengan struktur global macro inilah yang akan banyak dibahas dalam buku ini.

Struktur Local Macro

Suatu local macro terdiri dari baris-baris diantara kata: Berikut sruktur local macro:

```
macro
template
deklarasi peubah
default badan macro
endmacro
```

Macro dan Endmacro

Seperti halnya global macro, local macro juga diawali dan diakhiri dengan command MACRO dan ENDMACRO.

Template

Nama *template* untuk local macro sedikit berbeda dengan global macro karena terdapat tambahan argumen setelah nama macro. Nama *template* ini nantinya digunakan untuk memanggil local macro dan argumen yang menyertai layaknya menjalankan command/subcommand di Minitab.

Deklarasi Variabel

Variabel adalah nama yang mewakili suatu elemen data seperti **LK** untuk jenis kelamin, t4 lahir untuk tempat lahir, nama untuk nama dan lain sebagainya. Dalam memberi nama varibel ada aturan tertentu yang harus diikuti antara lain:

1. Harus dimulai dengan abjad, tidak boleh dengan angka

2. Apabila lebih dari satu kata tidak boleh menggunakan spasi
3. Sebaiknya memilih arti yang sesuai dengan elemen data
4. Sebaiknya variabel yang digunakan tidak terlalu panjang

Ada 3 jenis variabel yaitu [1][4]:

Mconstant

Digunakan untuk mendeklarasikan variabel yang bertipe konstant atau tunggal.

Mcolumn

Digunakan untuk mendeklarasikan variabel yang bertipe kolom atau vektor.

Mmatrix

Digunakan untuk mendeklarasikan variabel yang bertipe matriks.

3. METODE

Artikel ini disusun dengan beberapa langkah berikut:

Penjelasan operator dasar dalam makro minitab	Tahap pertama
Penejelasan logika setiap statement	Tahap Kedua
Studi kasus	Tahap ketiga

4. PEMBAHASAN

Operator Aritmatika

Operator aritmetika yang dapat digunakan yaitu:

**	RAISE OF POWER	Pangkat
*	MULT	Perkalian
/	DEVIDE	Pembagian
+	ADD	Penjumlahan
-	SUBSTRAK	Pengurangan
~=	NE	Tidak sama dengan
<	LT	Kurang dari
>	GT	Lebih dari
<=	LE	Kurang kecil sama dengan
>=	GT	Lebih besar sama dengan
=	EQ	Sama dengan

Program 1 Operasi aritmatika

```
Macro
aritm c1 c2
mcolumn c1 c2
mconstant k1 k2 k3 k4 k5 k6
let k1=mean(c1)
let k2=mean(c2)
let k3=k1+k2
let k4=k1-k2
let k5=k1*k2
let k6=k1/k1
print k1 k2 k3 k4 k5 k6
endmacro #akhir dari macro
```

Simpan dengan nama aritm.mac. jalankan macro di atas dengan cara mengetikkan di session window %d:aritma c1 c2 (%d jika file di simpan di storage d, aritm nama file, c1 dan c2 kolom yang akan di analisis)

Pada program 1. Baris pertama dan baris terakhir sebagai awal dan akhir macro. Baris kedua, aritm mewakili nama template program dengan dua variabel yaitu c1 dan c2. Selanjutnya, digunakan dua tipe variabel yaitu mcolumn (c1, c2) dan mconstant (k1, k2, k3, k4, k5, k6). Adapun let berfungsi untuk memberikan penugasan pada setiap variabel. Adapun variabel k1 untuk rata-rata c1, k2 untuk rata-rata c2 serta k3, k4, k5 dan k6 berturut-turut merupakan penjumlahan, pengurangan dan perkalian dari k1 dan k2.

Operator Logika

Operator logika adalah adalah operator yang digunakan untuk memberi nilai pada hasil perbandingan, atau kombinasi perbandingan. Adapun operator logika yang di gunakan dalam minitab [2]:

	OR	Atau
&	And	Dan
~	NOT	Tidak
%	Mod	Sisa pembagian

Program 2 Penggunaan Operator Logika

```
macro
opelogika x
```

```
mconstant n
mcolumn x
let n=x
if mod(n,2)=0
    note 'n adalah bilangan genap'
else
    note 'n adalah bilangan ganjil'
endif
endmacro
```

Terdapat dua variabel yang digunakan dalam program 2 yaitu constant untuk n dan column untuk x. Adapun outpunya terdapat dua alternatif, yaitu jika n dibagi 2 dan bersisa nola maka output programnya n adalah bilangan genap. Tetapi jika tidak maka outputnya 'n adalah bilangan ganjil'.

Looping/Perulangan

Perulangan merupakan sebuah konsep yang penting dalam pemrograman. Dengan struktur pengulangan, program dapat berjalan beberapa kali sesuai inisialisasi, jumlah iterasi dan kondisi berhenti yang ditentukan [6]. Untuk mengetahui konsep ini, perhatikan program berikut:

Program 3 Looping

```
macro
loop
mfree
    note saya suka minitab
    note saya suka minitab
    note saya suka minitab
note saya suka minitab
endmacro
```

Maksud dari program di atas adalah untuk menampilkan teks “saya suka minitab” sebanyak empat kali. Jika ditinjau dari hasilnya, program di atas memang benar, tetapi secara algoritma hal tersebut jelas salah karena tidak efisien, baik dilihat dari segi waktu maupun besarnya ukuran program yang dihasilkan. Mungkin untuk sembilan kali perulangan kita belum melihat bahwa program tersebut tidak efisien, namun bagaimana jika kita ingin menampilkan teks di atas sebanyak 1000 kali? Jika kita menggunakan cara di atas, tentu waktu dan

tenaga kita akan terbuang sia-sia. Maka dari itu kita harus menggunakan struktur pengulangan. Namun untuk melakukan perulangan statemen, tentu harus mengenal dengan baik aturan-aturan yang terdapat di dalamnya sehingga kita dapat terhindar dari kesalahan-kesalahan yang tidak diinginkan.

Struktur instruksi perulangan pada dasarnya terdiri atas:

- Kondisi perulangan: suatu kondisi yang harus dipenuhi agar perulangan dapat terjadi.
- Badan (body) perulangan: deretan instruksi yang akan diulang-ulang pelaksanaannya.
- Pencacah (counter) perulangan: suatu variabel yang nilainya harus berubah agar perulangan dapat terjadi dan pada akhirnya membatasi jumlah perulangan yang dilaksanakan,

Do – Enddo

Struktur pengulangan jenis ini digunakan untuk melakukan perulangan yang telah diketahui jumlah perulangannya. Perhatikan program berikut

Program 4 Penggunaan Do-Enddo

```
macro
product y
mconstant n i product
mcolumn y
let n = count(y)
let product = 1
do i=1 : n
let product=product*y(i)
enddo
print product
endmacro
```

While – Endwhile

Pada penggunaan struktur while, digunakan untuk melakukan perulangan yang tidak diketahui jumlah perulangannya. Sehingga statemen akan terus diulang sampai syarat yang diberikan tidak lagi terpenuhi.

```
while logical expression
```

```
statement -----
statement -----
endwhile
```

Program 5 penggunaan While-Endwhile

```
macro
bil c1
mcolumn c1
mconstant bil i
let bil=27
let i=1
while(bil<100)
let c1(i)= bil
let bil=bil+27
let i=i+1
endwhile
endmacro
```

Program 5 bertujuan untuk menampilkan bilangan kelipatan 27 yang kurang dari 100. Terdapat 3 variabel yang digunakan yaitu *bil*, *i* dan *c1*. Semua nilai dengan kelipatan 27 dimasukkan pada variabel *bil*. Setiap nilai variabel *bil* yang dihasilkan dalam perulangan selanjutnya disimpan pada variabel *c1* untuk ditampilkan pada worksheet minitab. Sedangkan untuk perintah while melakukan perulangan selama *bil* kurang dari 100.

Control Statemen

Penyeleksian kondisi digunakan untuk mengarahkan perjalanan suatu proses. Penyeleksian kondisi dapat diibaratkan sebagai katup atau kran yang mengatur jalannya air. Bila katup terbuka maka air akan mengalir dan sebaliknya bila katup tertutup air tidak akan mengalir atau akan mengalir melalui tempat lain. Seleksi kondisi adalah proses penentuan langkah berikutnya berdasarkan proses yang terjadi sebelumnya. Seleksi kondisi ini sangat penting dalam pemrograman sebab dengan adanya seleksi kondisi, program dapat menentukan proses apa yang harus dilakukan selanjutnya berdasarkan keadaan sebelumnya. Sehingga nampak seolah olah program dapat berpikir dan mengambil keputusan. Disinilah letak kekurangan komputer yaitu tidak mampu

berpikir sendiri, semua hal yang dilakukan adalah berdasarkan perintah. Control statemen dapat membuat lebih flexibel dan *powerful*, hal ini Karena mengijjinkan programmer [2].

Struktur IF terbagi menjadi tiga bagian antara lain

1. Struktur if (satu kondisi)
- Bentuk umum dari struktur if:
 If (kondisi)
 Statemen 1
 Statemen 2
 Endif

Program 6 penggunaan if

```
Macro
pemilihan_if
mconstant k1
let k1=10
if k1>0
note 'k1 merupakan bilangan
note asli'
endif
endmacro
```

Program 6 diberikan contoh penggunaan if dengan syarat jika variabel k1 yang bertipe constant lebih dari nol maka akan muncul di layar k1 merupakan bilangan asli.

2. Struktur IF-ELSE (satu kondisi)
- Pada struktur ini terdapat sebuah statemen khusus yang berguna untuk mengatasi kejadian apabila kondisi yang didefinisikan tersebut tidak terpenuhi. Bentuk umumnya adalah:
 If(kondisi)
 Statemen
 Else
 Statemen
 Endif

Program 7 Penggunaan if-else

```
Macro
perbandingan c1 c2
mcolumn c1 c2
name c1 'k1'
name c2 'k2'
Let c1 = 10
Let c2 = 15
```

```
If c1 < c2
note k1 'lebih kecil dari' k2
Else
note k2 'lebih besar dari' k1
Endif
endmacro
```

3. Struktur IF-ELSEIF (dua kondisi)
 Struktur ini merupakan perluasan dari IF-ELSE yakni menyisipkan (menmbahkan) satu atau lebih kondisi kedalamnya. Bentuk umum dari struktur percabangan ini yaitu:

```
If(kondisi 1)
Statemen 1
Statemen 2
.....
Elseif(kondisi 2)
Statemen 1
Statemen 2
.....
Else
Statemen 1
Statemen 2
.....
Endif
```

Memanggil Macro

Untuk memamnggil macro yang dibuat dalam file ASCCI. Kita harus berada dalam session.

- Pilih menu editor
- Pilih enable command
- Tulis % diikuti nama macro

Contoh:
 %D:\nama_forlnder\rata

Studi Kasus

Pada artike ini, akan dilakukan penggambaran secara umum tentang Produksi Kayu Bulat Oleh Perusahaan Hak Pengusahaan Hutan Menurut Jenis Kayu, 2004-2015. Sumber data Badan pusat statsitik Indonesia. Selanjutnya, dibuat macro minitab (Appendix A) untuk menggambarkan hasil produksi kayu bulat khususnya kayu aghatis. Pada program yang dibuat hanya ada dua jenis data yang digunakan yaitu constant dan column. Konstant terdiri atas jumx ratax varx jumvarx max min std i a letak t j k n d sisa r s p z, sedangkan column terdiri x kuartil

desil sisanya letaknya bulat
prstl urutan.

Pada badan macro, terdapat beberapa bagian berkaitan dengan statistic deskriptif yaitu: Pertama penentuan nilai rata-rata dan variansi.

```
1. let jumx=0
2. let jumvarx=0
3. let n=count(x)
4. do i=1:n
5. let jumx=jumx+x(i)
6. enddo
7. let ratax=jumx/n
8. do i=1:n
9. let jumvarx=jumvarx+((x(i)-ratax)**2)
10. enddo
11. let varx=jumvarx/(n-1)
12. let std=sqrt(varx)
```

Penentuan nilai rata-rata dan variansi dilakukan dengan memberikan nilai awal penjumlahan (identitas penjumlah 0) seperti pada baris pertama dan kedua, (perintah `let` digunakan untuk mendeklarasikan setiap variabel yang digunakan). Langkah berikutnya menentukan banyaknya data dengan menggunakan perintah `count`. Setelah itu menjumlahkan setiap item data dengan menggunakan perulangan `do-enddo` sebanyak n kali dengan `let jumx=jumx+x(i)`. Hal yang sama juga dilakukan untuk menentukan nilai jumlah nilai variansinya seperti pada baris ke-9 dan ke-10. Selanjutnya, baris ke-11 menentukan nilai variansi dan standar deviasi.

Bagian kedua penentuan nilai maximum dan minimum. Berikut ini diberikan proses penentuan maximum dan minimum:

```
1. let max=x(1)
2. let min=x(1)
3. do i=1:n
4. if x(i)>max
5. let max=x(i)
6. endif
7. if x(i)<min
8. let min=x(i)
9. endif
10. enddo
```

Untuk menentukan nilai maksimum dan minimum, dilakukan dengan membandingkan setiap data (baris ke-4) dimana data yang pertama (baris 1 dan 2) diasumsikan sebagai nilai maximum, minimum. Jika terdapat nilai data yang kurang atau lebih dari data pertama, maka data tersebut diambil sebagai data minimum (baris ke-7) atau maksimum (4).

Bagian ke-3 penentuan nilai kuartil. Untuk menentukan nilai kuartil, langkah pertama dilakukan adalah dengan mengurutkan data dari terkecil ke terbesar,

```
1. do i=1:n
2. do j=i+1:n
3. if x(i)>x(j)
4. let a=x(i)
5. let x(i)=x(j)
6. let x(j)=a
7. endif
8. enddo
9. enddo
```

pada baris 3, membandingkan setiap data dari data pertama sampai data ke- n . jika data yang memenuhi syarat $x(i) > x(j)$, dimana $i=1..n$, $j=2..n$ maka dilakukan penukaran posisi seperti pada baris ke-4,5,6. Setelah data terurut, selanjutnya menentukan nilai kuartil data.

```
1. do i=1:3
2. let k=i*(n+1)/4
3. let t=floor(k,0)
4. let sisa=k-t
5. let r=t+1
6. let
kuartil(i)=x(t)+sisa*(x(r)-x(t))
7. enddo
```

Baris pertama, perintah perulangan dari $i=1$ sampai 3, hal ini Karena kuartil terdiri dari 3 jenis yaitu kuartil satu, kuartil dua dan kuartil 3. Penentuan letak kuartil dilakukan pada baris ke-2, sedangkan baris ke-6 merupakan proses penentuan nilai kuartil, dimana $sisa = k - t$, dengan t merupakan pembulatan ke-bawah dari letak kuartil k .

Bagian Ke-4 Macro Penentuan Desil. Desil membagi data dengan 10 bagian. Pada dasarnya proses penentuan desil sama dengan proses

penentuan kuartil, tinggal baris pertama angka 3 diganti dengan angka 10 (banyaknya desial ada 10) dan baris ke-2 pembagi 4 diganti dengan 10. Demikian hanya dengan persentil, angka 3 diganti dengan 100 dan angka 4 diganti dengan 100. Lebih lengkapnya bias diperhatikan pada appendix A.

Selanjutnya, dilakukan analisis berdasarkan output program. Setelah program dijalankan diperoleh rata-rata produksi kayu aghatis dari tahun 2004 sampai tahun 2015 sebesar 11161 batang dalam setiap tahunnya. Sedangkan besar keragaman mencapai 143945712. Produksi paling sedikit sebanyak 453 yaitu pada tahun 2013 dan 2014 sedangkan yang paling banyak sebesar 32134 pada tahun 2004.

Selanjutnya, hasil produksi dipandang berdasarkan ukuran gejala pusat. Jika hasil produksi dibagi menjadi 4 kuartal. Maka nilai pertengahan untuk kuartal pertama sebesar 1752 kuartal kedua (median) hasil produksi sebanyak 5943 sedangkan untuk kuartal ketiga sebesar 24129. Sedangkan jika dibagi menjadi 10 atau 100 bagian maka hasilnya dapat dilihat pada appendix B. Sedangkan gambaran hasil produksi tahun 2004-2015, secara umum terjadi penurunan hasil produksi. Hanya pada tahun 2004, 2005, 2007 dan 2008 memiliki produksi di atas rata-rata.

5. KESIMPULAN

Berdasarkan hasil pembahasan, maka disimpulkan bahwa, macro minitab dapat dijadikan suatu alternative untuk mengolahan dan menganalisis data untuk memeberikan penjelasan dan kesimpulan terhadap permasalahan yang berkaitan dengan statistik ataupun matematik.

6. DAFTAR PUSTAKA

- [1] Iriawan Nur dan Astuti Puji Septi. *Mengolah Data Statistik dengan Mudah Menggunakan Minitab 14*. Yogyakarta: Penerbit Andi. 2006
- [2] Minitab User's Guide 1: Data, Graphics, and macros

- [3] Muhammad Rifqi S. *Pelatihan Macro minitab & excel statistics computer course*. 2009.
- [4] Staf Jurusan Statistika. *Panduan Pengolahan Data dengan paket program minitab windows edisi kedua*. Bogor: Institut Petanian Bogor.
- [5] Suparto. *Bahasa Pemrograman Untuk Sekolah Menengah Kejuruan*. 2008.
- [6] Politeknik Telkom Bandung. *Algoritma dan Pemrograman*. 2009

APENDIKS A**MACRO MINITAB**

```

macro
rata_rata x
mconstant  jumx ratax varx
mconstant  jumvarx  max min std
mconstant  i a letak t j k n d sisa
mconstant  r s p z
mcolumn x kuartil desil
mcolumn sisanya letaknya
mcolumn bulat prstl urutan
TSPlot x
#UKURAN PENYEBARAN
#RATA-RATA DAN VARIANSI
let jumx=0
let jumvarx=0
let n=count(x)
do i=1:n
  let jumx=jumx+x(i)
enddo
let ratax=jumx/n
do i=1:n
let      jumvarx=jumvarx+((x(i) -
  ratax)**2)
enddo
let varx=jumvarx/(n-1)
let std=sqrt(varx)
print ratax varx  std

#menentukan max dan min
let max=x(1)
let min=x(1)
do i=1:n
  if x(i)>max
    let max=x(i)
  endif
if x(i)<min
  let min=x(i)
endif
enddo
print max min

#UKURAN GEJALA PUSAT
#QUARTIL
do i=1:n
do j=2:n
if j>i and x(i)>x(j)
let a=x(i)
let x(i)=x(j)
let x(j)=a
endif
enddo
do z=1:3
let k=i*(n+1)/4
let t=floor(k,0)
let sisa=k-t
let r=t+1
let kuartil(i)=x(t)+sisa*(x(r)-
  x(t))
enddo
print  urutan  letaknya  bulat
  sisanya kuartil

#Desil
do z=1:9
let r=n+1
let d=(z*r)/10
let p=floor(d,0)
let s=p+1
let sisa=d-p
let desil  (z)=x(p)+sisa*(x(s)-
  x(p))
enddo
print desil

#PERSENTIL
do z=1:99
let r=n+1
let d=(z*r)/100
let p=floor(d,0)
let s=p+1
let sisa=d-p

#nilai persentil
let      prstl(z)=x(p)+sisa*(x(s)-
  x(p))
enddo
print urutan letaknya bulat
print sisanya prstl

endmacro

```

APENDIX B

OUTPUT MACRO MINITAB

Data Display

ratax 11161.7
varx 143945712
std 11997.7

Data Display

max 32134.0
min 453.000

Data Display

Data

Row	urutan	letaknya	bulat	sisanya	kuartil
1	1	3.25	3	0.25	1752.3
2	2	6.50	6	0.50	5943.5
3	3	9.75	9	0.75	24129.3

Data Display

Data

Row	urutan	letaknya	bulat	sisanya	desil
1	1	1.3	1	0.3	453.0
2	2	2.6	2	0.6	1148.4
3	3	3.9	3	0.9	2116.9
4	4	5.2	5	0.2	4936.2
5	5	6.5	6	0.5	5943.5
6	6	7.8	7	0.8	6310.8
7	7	9.1	9	0.1	18922.1
8	8	10.4	10	0.4	27634.4
9	9	11.7	11	0.7	31460.2

Data Display

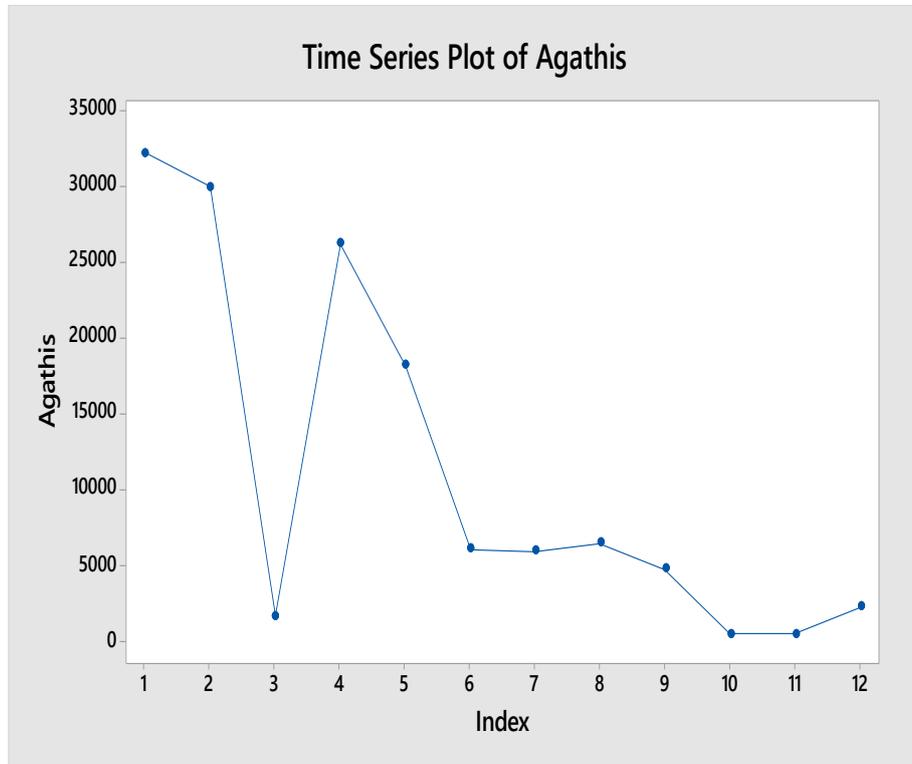
Data

Row	sisanya	prstl
1	0.13	*
2	0.26	*
3	0.39	*
4	0.52	*
5	0.65	*
6	0.78	*
7	0.91	*
8	0.04	453.0
9	0.17	453.0
10	0.30	453.0
11	0.43	453.0
12	0.56	453.0
13	0.69	453.0
14	0.82	453.0
15	0.95	453.0
16	0.08	545.7
17	0.21	696.4
18	0.34	847.1
19	0.47	997.7
20	0.60	1148.4
21	0.73	1299.1
22	0.86	1449.7
23	0.99	1600.4
24	0.12	1679.3
25	0.25	1752.3
26	0.38	1825.2
27	0.51	1898.1
28	0.64	1971.0
29	0.77	2044.0
30	0.90	2116.9
31	0.03	2249.0
32	0.16	2578.4

33	0.29	2907.9	68	0.84	16242.4
34	0.42	3237.3	69	0.97	17768.8
35	0.55	3566.7	70	0.10	18922.1
36	0.68	3896.1	71	0.23	19963.5
37	0.81	4225.5	72	0.36	21005.0
38	0.94	4555.0	73	0.49	22046.4
39	0.07	4787.2	74	0.62	23087.8
40	0.20	4936.2	75	0.75	24129.3
41	0.33	5085.2	76	0.88	25170.7
42	0.46	5234.2	77	0.01	26169.6
43	0.59	5383.1	78	0.14	26657.8
44	0.72	5532.1	79	0.27	27146.1
45	0.85	5681.1	80	0.40	27634.4
46	0.98	5830.1	81	0.53	28122.7
47	0.11	5872.9	82	0.66	28611.0
48	0.24	5896.4	83	0.79	29099.2
49	0.37	5920.0	84	0.92	29587.5
50	0.50	5943.5	85	0.05	30000.3
51	0.63	5967.0	86	0.18	30292.3
52	0.76	5990.6	87	0.31	30584.3
53	0.89	6014.1	88	0.44	30876.2
54	0.02	6040.9	89	0.57	31168.2
55	0.15	6085.9	90	0.70	31460.2
56	0.28	6130.9	91	0.83	31752.2
57	0.41	6175.9	92	0.96	32044.2
58	0.54	6220.8	93	0.09	*
59	0.67	6265.8	94	0.22	*
60	0.80	6310.8	95	0.35	*
61	0.93	6355.8	96	0.48	*
62	0.06	7084.5	97	0.61	*
63	0.19	8610.8	98	0.74	*
64	0.32	10137.1	99	0.87	*
65	0.45	11663.4			
66	0.58	13189.8			
67	0.71	14716.1			

APENDIX C

Grafik hasil produksi kayu aghatis



ANALISIS TINGKAT KEPUASAN PENGGUNA LABORATORIUM KOMPUTER JURUSAN MATEMATIKA	1 - 8
<i>wahidah alwi</i>	
MODEL MIXTURE SURVIVAL PADA KETAHANAN HIDUP PENDERITA TUMOR DENGAN PENDEKATAN REGRESI COX PROPORTIONAL HAZARD.....	9 - 13
<i>Adnan Sauddin, Indo Upe</i>	
APLIKASI FUZZY INTEGER TRANSPORTATION DALAM OPTIMASI BIAYA DISTRIBUSI SEPEDA MOTOR PADA PT. NUSANTARA SURYA SAKTI	14 - 23
<i>Risnawati Ibtnas, Hijaz K Musgami</i>	
JUMLAH BESARAN KLAIM DENGAN METODE REKURSIF ($N \sim \text{Poisson}(\gamma)$) ...	24 - 29
<i>Faihatuz</i>	
Solusi Persamaan Diferensial Biasa dengan Metode Runge-Kutta Orde Lima	30 - 36
<i>Fardinah</i>	
Inventory dan Transportasi pada Jaringan Berkendala Menggunakan Simultaneous Optimization Method	37 - 45
<i>Wahyu Sri Utami</i>	
■	
PERBANDINGAN SOLUSI NUMERIK INTEGRAL LIPAT DUA PADA FUNGSI ALJABAR DENGAN METODE ROMBERG DAN SIMULASI MONTE CARLO ...	46 - 57
<i>Ermawati, Puji rahayu, Faihatuz</i>	
TUTORIAL SOFTWARE: ANALISIS DATA MENGGUNAKAN PEMROGRAMAN MAKRO MINITAB	58 -68
<i>Muh. Irwan</i>	
MEMODELKAN FAKTOR-FAKTOR YANG MEMPENGARUHI GIZI BURUK BALITA DENGAN METODE MULTIVARIATE ADAPTIVE REGRESSION SPLINE (MARS)	69 - 74
<i>Irwan Kasse, Mawar Wardani</i>	
PERBANDINGAN METODE ADM (Adomian Decomposition Method) DAN METODE FTCS (Forward Time Central Space) DALAM MENYELESAIKAN MASALAH ORDE n PADA PERSAMAAN DIFERENSIAL PARSIAL	75 - 80
<i>Muhammad Kaprawi</i>	
SOLUSI INTEGRASI NUMERIK DENGAN METODE SIMPSON (SIMPSON'S RULE) PADA TRANSFORMASI HANKEL	81 - 86
<i>wahidah alwi, Ermawati Ermawati, Nur Syamsi</i>	

