



Analisis Angka Kematian Bayi dan Angka Harapan Hidup Kab/Kota di Jawa Timur dengan Menggunakan Library Gregmisc

Oleh : *Unung Istopo H (unung@enciety.com)*

Untuk analisis data ini kita menggunakan data di <http://www.enciety.com/edm/manajemen/>.

Berikut ini script yang dapat kita gunakan :

```
# Mengambil data dari enciety Data Manajemen
read.csv("http://www.enciety.com/edm/manajemen
/data/AKB%20dan%20AHH%20Kab%20Kota%201999%20-
%202001%20-%202003%20Jawa%20Timur.csv") ->
data.AKB.AHH

require(gregmisc)

# Misal untuk analisis tahun 2003

x <-
data.frame(data.AKB.AHH[1:37,c(2,5,8)],row.nam
es=1)

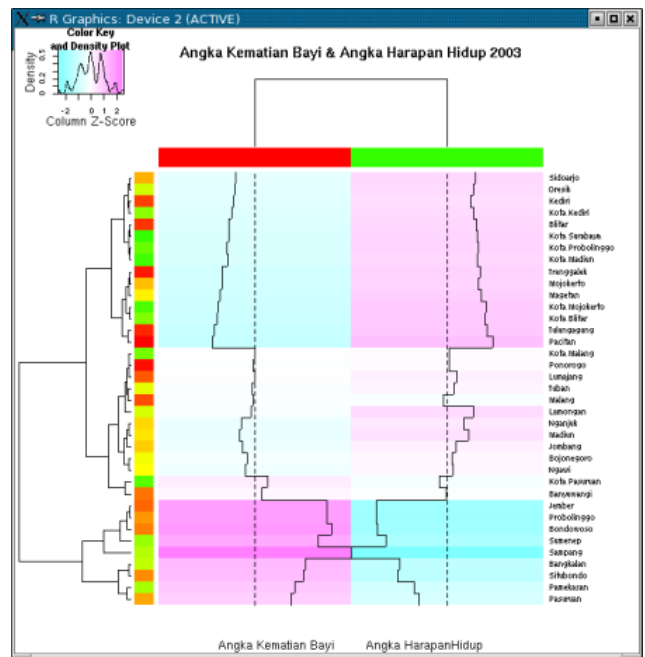
as.matrix(x) -> x

rc <- rainbow(nrow(x), start=0, end=.3)
cc <- rainbow(ncol(x), start=0, end=.3)

hv <- heatmap.2(x, col = cm.colors(256),
scale="column", RowSideColors = rc,
ColSideColors = cc, margin=c(5,10), xlab =
"Angka Kematian Bayi      Angka
HarapanHidup", ylab= "", main = "Angka
Kematian Bayi & Angka Harapan Hidup 2003",
tracecol="black", density="density")

str(hv) # the two re-ordering index vectors
```

Hasil Analisisnya adalah sebagai berikut :



Analisis dendrogram dua sisi ini menunjukkan cluster dari cases sekaligus cluster dari variabel pembentuknya.

Bagaimana melakukan instalasi R di Linux

Untuk melakukan instalasi R di Linux, terlebih dahulu download R versi terbaru dari <http://cran.r-project.org>, kemudian ekstrak file tersebut pada direktori tertentu, misalnya /home/data/master/R setelah itu panggil aplikasi console dan arahkan ke top direktori.

Jika sudah maka ketikkan perintah berikut ini :

```
./configure
make
```

Kemudian lakukan pengecekan dari sistem yang sudah dibangun secara benar, melalui :

```
make check
```

Untuk membangun manual, lanjutkan dengan :

```
make dvi
make pdf
make info
```

Lanjutkan dengan instalasi melalui root, dengan menggunakan perintah :

```
make install
make install -dvi
make install -pdf
make install -info
```

Instalasi ini dilakukan di /usr/local/. Untuk melakukan option instalasi lain ikuti *installation and Administration Manual*.

Plot Time Series : Message History Parameter YAHOO! Groups

Oleh : Unung Istopo H

Kita seringkali ingin mengetahui bagaimana time series messages dalam komunitas milis yang kita ikuti. Namun terkadang, model message History Parameter, seperti dalam Gambar 1, sulit bagi kita untuk melakukan analisis. Dalam script sederhana kali ini, kita akan mencoba menggambarkan plot time series untuk data yang berbentuk matriks. Sebenarnya kita bisa menggunakan fungsi-fungsi R Time Series, namun untuk langkah-langkah ini kita akan menggunakan script dasar R, sekaligus untuk mengenalkan bagaimana menggunakan loop, abline dan fungsi dasar lainnya.

Gambar 1. Data Milis Parameter YAHOO! groups Message History

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2005	110	65	34	49	55	21						
2004	158	99	130	61	45	79	84	82	86	69	44	51
2003	163	76	135	180	99	124	150	97	83	137	138	182
2002	36	58	28	33	31	8	55	85	71	75	129	62
2001											19	31

Snapshot : 17 Juni 2005 10:53 WIB

Langkah-Langkah membuat plot time series untuk message history milis <http://www.yahogroups.com/> dengan kasus milis Parameter yang kita ambil pada tanggal 17 Juni 2005, adalah sebagai berikut :

1. Mengambil data *message history* dari halaman <http://groups.yahoo.com/group/Parameter/>.

Ada beberapa metode untuk mengambil data dari halaman ini, yaitu dengan menggunakan metode RegEx, untuk mengambil data dalam pola HTML tertentu atau cara yang mudah dan simple yaitu dengan copy clipboard untuk data message history.

Untuk script kali ini kita memilih copy clipboard, dengan cara seperti artikel sebelumnya (<http://www.enciety.com/community/R/?p=37>).

```
read.delim(pipe("xclip -o")) -> data.parameter
```

2. Kemudian kita melakukan manipulasi data matriks diatas untuk membuat data series, berikut ini script yang dapat digunakan :

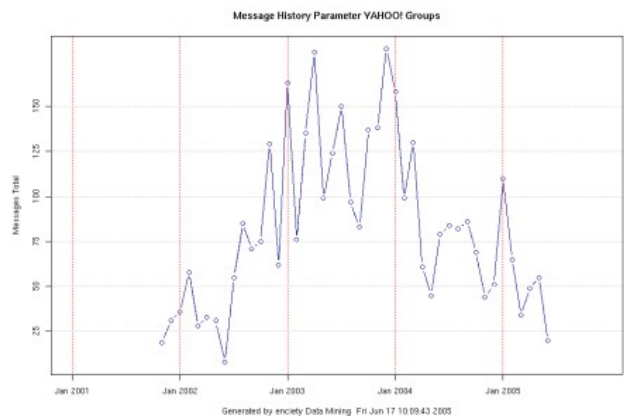
```
data.new <- c(1:60)
for (i in 5:1) {
  for (j in 1:12){
    data.parameter[i,j] -> data.new[j+(12*(5-i))]
  }
}
```

3. Selanjutnya, kita dapat mulai melakukan plot, menambahkan abline untuk mempercantik tampilan.

```
now <- date()
plot(data.new,type="b",axes=F,xlab=paste("Gene
rated by enciety Data Mining
",now),ylab="Messages Total",main="Message
History Parameter YAHOO!
Groups",col="darkblue")
axis(2)
abline(v=c(1,13,25,37,49),lty=3,col="red")
abline(h=seq(25,150,25),lty=3,col="grey")
axis(2,at=seq(25,150,25),seq(25,150,25))
axis(1,at=c(1,13,25,37,49),c("Jan 2001","Jan
2002","Jan 2003","Jan 2004","Jan 2005"))
#text(c(1:60),data.new+3,data.new)
box()
```

4. Setelah kita jalankan script tersebut di R maka kita akan dapatkan hasil seperti Gambar 2.

Gambar 2. Plot Time Series Milis Parameter YAHOO! groups Message History



Snapshot : 17 Juni 2005 11:19 WIB

Bubble Plot Mapping : Menggambarkan Distribusi Data di R dengan Bubble Symbol

Oleh : Unung Istopo H

Menggambarkan distribusi data, khususnya untuk menggambarkan keragaman data antar Kabupaten / Kota dapat kita lakukan dengan menggunakan simbol Bubble Plot. Di mana besar kecilnya data dapat dibedakan melalui radius masing-masing nilai tiap Kabupaten / Kota.

Untuk kasus berikut ini, kita akan mencoba menggambarkan distribusi Angka Buta Huruf Propinsi Jawa Timur pada Tahun 1998. Script yang digunakan adalah sebagai berikut:

```
## Penggunaan bubble plot peta Jawa Timur
## Oleh : Unung Istopo H
## 20 Juni 2005

# Load dasar Map Jatim
source("/home/unung/work/MapSource/LibraryRequire.R")

library(maptools)

read.shape("/home/unung/work/MapSource/jatim.shp") -> jatim

read.csv("http://www.encyety.com/edm/manajemen/data/DATA%20ANGKA%20BUTA%20HURUF%20PENDUDUK%20UMUR%2010TH%20KE%20ATAS-BPS-TH2003-%20(Persen)%20-JATIM.csv") -> data

data[1:37,3] -> data
cbind(urut[,3:4],data) -> data.new
data.new[order(data.new[,2]),] -> data

now <- date()

plot(jatim,fg="grey89",main=paste("Bubble Plot Mapping", "\n", "Distribusi Prosentase Angka Buta Huruf di Jawa Timur 1998"),xlab=paste("Generated by Enciety Data Mining -",now),ylab="")

par(new=T)

x <- c(1:37)
y <- c(1:37)

for (i in 1:37) {
  (jatim$Shapes[[i]]$bbox[1]+jatim$Shapes[[i]]$bbox[3])/2 -> x[i]
  (jatim$Shapes[[i]]$bbox[2]+jatim$Shapes[[i]]$bbox[4])/2 -> y[i]
}
```

```
for (i in 1:37){
  symbols(x[i],y[i],circles=data[i,3]/275,bg="red",fg="yellow",xlim=c(attributes(jatim$Shapes)$minbb[1],attributes(jatim$Shapes)$maxbb[1]),ylim=c(attributes(jatim$Shapes)$minbb[2],attributes(jatim$Shapes)$maxbb[2]),inches=F,add=T)
}

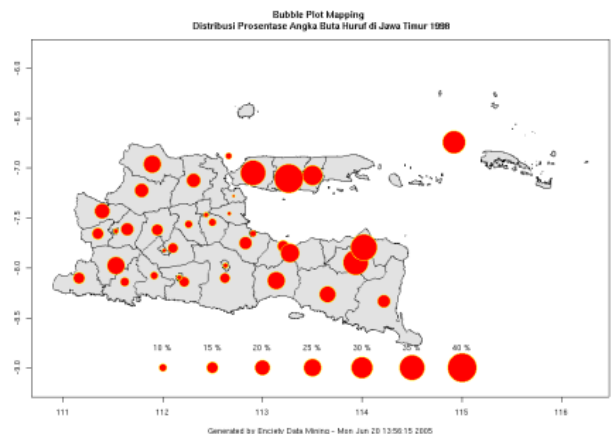
par(new=T)

legend.x <- c(112,112.5,113,113.5,114,114.5,115)
legend.y <- c(-9,-9,-9,-9,-9,-9,-9)
legend.data <- c(10,15,20,25,30,35,40)

for (i in 1:7){
  symbols(legend.x[i],legend.y[i],circles=legend.data[i]/275,bg="red",fg="yellow",xlim=c(attributes(jatim$Shapes)$minbb[1],attributes(jatim$Shapes)$maxbb[1]),ylim=c(attributes(jatim$Shapes)$minbb[2],attributes(jatim$Shapes)$maxbb[2]),inches=F,add=T)
}

text(legend.x,legend.y+0.2,c("10 %", "15 %", "20 %", "25 %", "30 %", "35 %", "40 %"))
```

Hasilnya adalah sebagai berikut :



Sumber Data : BPS, Generated by : Enciety Data Mining