Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

*Chapter 7*
*Cluster Analysis and Trees*

Analisis *Cluster* merupakan metode yang dikembangkan untuk menemukan gen yang membentuk kelompok. Metode ini didasarkan pada suatu fungsi *distance* dan algoritma sehingga beberapa *cluster* gen dapat ditemukan, yang biasanya menerapkan analisis *cluster linkage* tunggal. Adapun metode analisis *cluster k-means* yang dapat digunakan dengan menghasilkan *tree*, yang merepresentasikan kemiripan dan perbedaan gen. Koefisien korelasi juga digunakan untuk menyelidiki kemiripan suatu pasangan ekspresi gen.

## 7.1 *Distance*

Pada analisis *cluster*, konsep perhitungan jarak memiliki peran penting, yang didefinisikan sebagai nilai mutlak dari ketidakmiripannya. Jarak *Euclidian* yang didefinisikan sebagai akar dari jumlah perbedaan kuadrat digunakan untuk situasi ketika suatu nilai ekspresi gen untuk beberapa pasien untuk menentukan vektor jarak dari ekspresi gen. Jarak E*uclidian* digunakan untuk menghitung jarak antara dua vektor.

* Menentukan jarak antara 2 vektor ekspresi gen menggunakan jarak Euclidian

$$d(\boldsymbol{a},\boldsymbol{b}) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}.$$

* Pengaplikasian pada pemrograman R

```
> #Cluster Analysis and Trees
> # Ex 3
> #To compute the Euclidian distance between two vectors
> a <- c(1,1); b <- c(4,5)
> sqrt(sum((a-b)^2))
[1] 5
> #ex 4
> # Distances between Cyclin gene expressions
> # BiocManager::install("multtest")
> library(multtest); data(golub)
> index <- grep("Cyclin",golub.gnames[,2])
> golub.gnames[index,2]
 [1] "CCND2 Cyclin D2"
 [2] "CDK2 Cyclin-dependent kinase 2"
 [3] "CCND3 Cyclin D3"
 [4] "CDKN1A Cyclin-dependent kinase inhibitor 1A (p21, Cip1)"
 [5] "CCNH Cyclin H"
 [6] "Cyclin-dependent kinase 4 (CDK4) gene"
 [7] "Cyclin G2 mRNA"
 [8] "Cyclin A1 mRNA"
 [9] "Cyclin-selective ubiquitin carrier protein mRNA"
[10] "CDK6 Cyclin-dependent kinase 6"
[11] "Cyclin G1 mRNA"
[12] "CCNF Cyclin F"
> dist.cyclin <- dist(golub[index,],method="euclidian")
> diam <- as.matrix(dist.cyclin)
> rownames(diam) <- colnames(diam) <- golub.gnames[index,3]
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

```
> diam[1:5,1:5]
          D13639_at M68520_at M92287_at U09579_at U11791_at
D13639_at  0.000000  8.821806  11.55349 10.056814  8.669112
M68520_at  8.821806  0.000000  11.70156  5.931260  2.934802
M92287_at 11.553494 11.701562   0.00000 11.991333 11.900558
U09579_at 10.056814  5.931260  11.99133  0.000000  5.698232
U11791_at  8.669112  2.934802  11.90056  5.698232  0.000000
> #Example 5
> # Finding the ten closest genes to a given one
> # BiocManager::install("genefilter")
> # BiocManager::install("Rcpp")
> #library("genefilter"); library("ALL"); data(ALL)
> closeto1389_at <- genefinder(ALL, "1389_at", 10, method = "euc")
> closeto1389_at[[1]]$indices
 [1]  2653  1096  6634  9255  6639 11402  9849  2274  8518 10736
> round(closeto1389_at[[1]]$dists,1)
 [1] 12.6 12.8 12.8 12.8 13.0 13.0 13.1 13.2 13.3 13.4
> featureNames(ALL)[closeto1389_at[[1]]$indices]
 [1] "32629_f_at" "1988_at"    "36571_at"   "39168_at"
 [5] "36576_at"   "41295_at"   "39756_g_at" "32254_at"
 [9] "38438_at"   "40635_at"
```

## 7.2 Dua Tipe Analisis *Cluster*

Analisis kluster tautan tunggal dapat diterapkan untuk mengeksplorasi grup dalam satu set ekspresi gen. Ketika kelompok hadir, analisis *cluster* k dapat berarti diterapkan dalam kombinasi dengan bootstrap untuk memperkirakan interval kepercayaanuntuk cluster *k-means.*

### 7.2.1 *Single Linkage*

Algoritma analisis *cluster single linkage:*

* Membuat *cluster* sebagai tiitk data

* Dua yang terdekat akan digabungkan menjadi satu

* Dua yang terdekat selanjutnya akan digabungkan juga menjadi satu

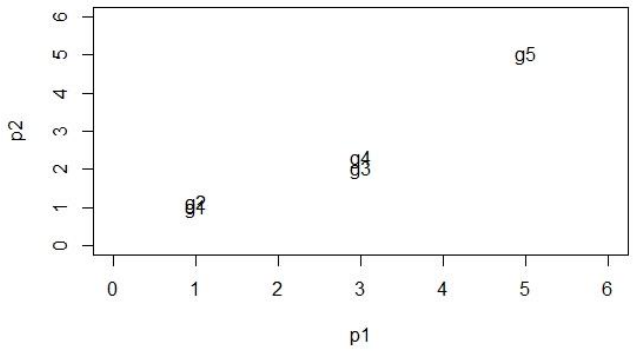
* Proses ini dilakukan secara berulang hingga semua poin menjadi satu *cluster*.

Contoh 1 Ilustrasi *cluster single linkage*

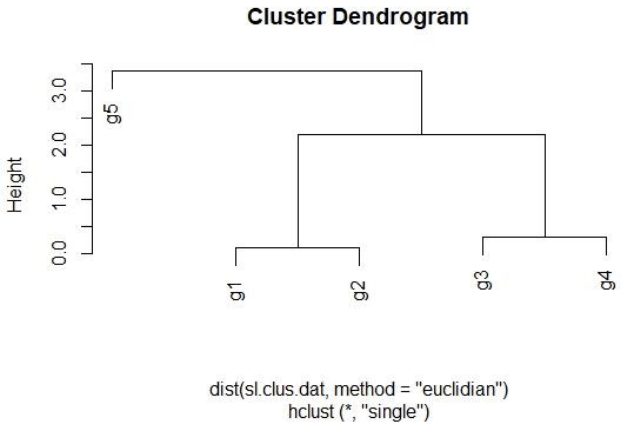
```
> #7.2 Cluster Analysis
> #Single Linkage
> # computes the the distances between the genes and performs a single
linkage cluster analysis
> #Ex 1
> names <- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
> sl.clus.dat <- matrix(c(1,1,1,1.1,3,2,3,2.3,5,5),ncol = 2,
+               byrow = TRUE,dimnames = names)
> # pict 7.1
> plot(sl.clus.dat,type="n", xlim=c(0,6), ylim=c(0,6))
> text(sl.clus.dat,labels=row.names(sl.clus.dat))
> print(dist(sl.clus.dat,method="euclidian"),digits=3)
     g1   g2   g3   g4
g2 0.10
g3 2.24 2.19
g4 2.39 2.33 0.30
g5 5.66 5.59 3.61 3.36
> # pict 7.2 cluster dendogram
> sl.out<-hclust(dist(sl.clus.dat,method="euclidian"),method="single")
> plot(sl.out)
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*



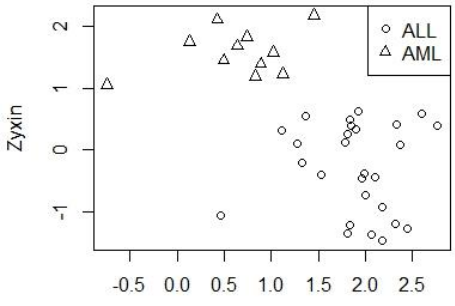

*Gambar 7 2 Plot 5 poin yang telah dicluster*

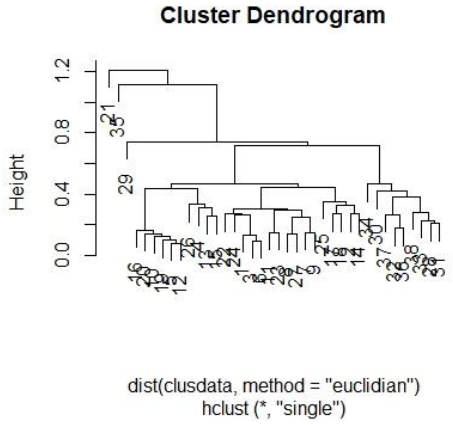



*Gambar 7 1 Tree of single linkage cluster anaylisis*

Contoh 2 Menghubungkan proses pembuatan data dengan pohon *cluster*

```
> #Example 2
> sl.out<-hclust(dist(rnorm(20,0,1),method="euclidian"),method="sing
le")
> #pict 7.3
> plot(sl.out)
> x <- c(rnorm(10,0,0.1),rnorm(10,3,0.5),rnorm(10,10,1.0))
> #pict 7.4
> plot(hclust(dist(x,method="euclidian"),method="single"))
> #pict 7.5
> plot(sl.out)
> #Example 3
> # Recall that the first twenty seven patients belong to ALL and the
 remaining eleven to AML and that we found earlier that the expressi
on values of the genes
> data(golub, package="multtest")
> clusdata <- data.frame(golub[1042,],golub[2124,])
> colnames(clusdata)<-c("CCND3 Cyclin D3","Zyxin")
> gol.fac <- factor(golub.cl,levels=0:1, labels= c("ALL","AML"))
> #pict 7.5
> plot(clusdata, pch=as.numeric(gol.fac))
> legend("topright",legend=c("ALL","AML"),pch=1:2)
> #pict 7.6
> # tree from single linkage cluster analysis
> plot(hclust(dist(clusdata,method="euclidian"),method="single"))
```



Fadhillah Putri Taha
H071171301

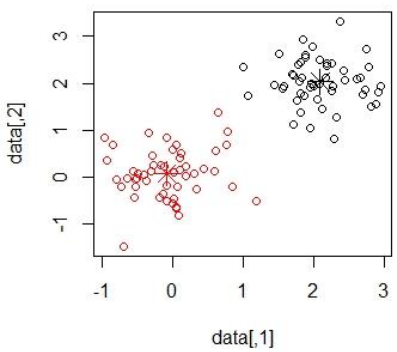
*Summary Cluster Analysis dan Trees*

### 7.2.2 *K-Means*

Analisis *clustering k-means* merupakan metode popular dalam Bioinformatika, yang dapat meminimalkan jumlah *cluster* dalam kuadrat atas *cluster k.*

- Membuat partisi
- Kelompok baru akan dihitung
- Mengaitkan setiap titik dengan *mean cluster* terdekatnya
- Proses ini terus berulang hingga mencapai konvergensi, dimana ketika titik data tidak lagi mengubah *cluster*.

```
> #K-means cluster analysis

> #ex 1
> # Relating a data generation process to k-means cluster analysis
> data <- rbind(matrix(rnorm(100,0,0.5), ncol = 2),
+     + matrix(rnorm(100,2,0.5), ncol = 2))
> cl <- kmeans(data, 2)
> # Specify the color of each data
> # cluster membership
> #pict 7.7
> plot(data, col = cl$cluster)
> points(cl$centers, col = 1:2, pch = 8, cex=2)
> # the sample means of potential clusters or the hypothesized pop
ulation means
> initial <- matrix(c(0,0,2,2), nrow = 2, ncol=2, byrow=TRUE)
> cl<- kmeans(data, initial, nstart = 10)
> #Calculate quantile
> n <- 100; nboot<-1000
> boot.cl <- matrix(0,nrow=nboot,ncol = 4)
> for (i in 1:nboot){
+   dat.star <- data[sample(1:n,replace=TRUE),]
+   cl <- kmeans(dat.star, initial, nstart = 10)
+   boot.cl[i,] <- c(cl$centers[1,],cl$centers[2,])
+ }
> quantile(boot.cl[,1],c(0.025,0.975))
      2.5%      97.5%
-0.1333764  0.1288199
> quantile(boot.cl[,2],c(0.025,0.975))
      2.5%      97.5%
-0.1053102  0.2230752
> quantile(boot.cl[,3],c(0.025,0.975))
    2.5%    97.5%
1.984643 2.274903
> quantile(boot.cl[,4],c(0.025,0.975))
    2.5%    97.5%
1.931078 2.205818
```



Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

Contoh 2

```
> #Ex 2

> data <- data.frame(golub[1042,],golub[2124,])
> colnames(data)<-c("CCND3 Cyclin D3","Zyxin")
> cl <- kmeans(data, 2,nstart = 10)
> #K-means clustering with 2 clusters of sizes 11, 27
> cl
K-means clustering with 2 clusters of sizes 11, 27

Cluster means:
  CCND3 Cyclin D3      Zyxin
1       0.6355909  1.5866682
2       1.8938826 -0.2947926

Clustering vector:
 [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1
1 1 1 1 1 1 1

Within cluster sum of squares by cluster:
[1]  4.733248 19.842225
 (between_SS / total_SS =  62.0 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "to
t.withinss" "betweenss"
[7] "size"         "iter"         "ifault"
> mean(data.frame(boot.cl))
[1] NA
Warning message:
In mean.default(data.frame(boot.cl)) :
  argument is not numeric or logical: returning NA
> quantile(boot.cl[,1],c(0.025,0.975))
 2.5% 97.5%
    0     0
> quantile(boot.cl[,2],c(0.025,0.975))
 2.5% 97.5%
    0     0
> quantile(boot.cl[,3],c(0.025,0.975))
 2.5% 97.5%
    0     0
> quantile(boot.cl[,4],c(0.025,0.975))
 2.5% 97.5%
    0     0
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

## 7.3 Koefisien Korelasi

Koefisien yang sering digunakan untuk menyatakan tingkat hubungan linier antara 2 nilai ekspresi gen disebut koefisien korelasi $\rho$.

```
> #Correlation Coefficient

> #ex 1
> #library(TeachingDemos)
> # run.cor.examp(1000)
> #ex 2
> # put.points.demo()
> #ex 3
> library(multtest); data(golub)
> x <- golub[2289,]; y <- golub[2430,]
> cor(x,y)
[1] 0.6376217
> cor.test(x,y)

        Pearson's product-moment correlation

data:  x and y
t = 4.9662, df = 36, p-value = 1.666e-05
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3993383 0.7952115
sample estimates:
      cor
0.6376217

> #Ex 4
> nboot <- 1000; boot.cor <- matrix(0,nrow=nboot,ncol = 1)
> data <- matrix(c(x,y),ncol=2,byrow=FALSE)
> for (i in 1:nboot){
+   dat.star <- data[sample(1:nrow(data),replace=TRUE),]
+   boot.cor[i,] <- cor(dat.star)[2,1]}
> mean(boot.cor)
[1] 0.6529344
> quantile(boot.cor[,1],c(0.025,0.975))
     2.5%     97.5%
0.2406223 0.9199311
> #ex 5
> library(multtest); data(golub)
> corgol<- apply(golub, 1, function(x) cor(x,golub.cl))
> o <- order(corgol)
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

## 7.4 Analisis Komponen Utama

Analisis komponen utama adalah metode deskriptif untuk menganalisis ketergantungan dencies (korelasi) antara variabel. Jika ada beberapa nilai eigen besar, lalu ada banyak arah dalam data yang merangkum variasi paling penting di antara ekspresi gen. Maka mungkin digunakan untuk mengeksplorasi secara simultan visualisasi gen dua dimensi dan para pasien. Selain itu, dapat bermanfaat untuk mempelajari bobot vektor eigen karena ini dapat mengungkapkan struktur dalam data sebaliknya luput dari perhatian. Akhirnya, komponen utama mengandung lebih sedikit (ukuran-ment) kesalahan daripada variabel individu.

```
> # Principal Components Analysis
> #ex 1 - Compute correlation matrix
> Z <- matrix(c( 1.63, 1.22, -0.40, 0.79, 0.93, 0.97, -1.38,
+       -1.08, -0.17, -0.96, -0.61, -0.93), nrow=6, byrow=TRUE)

> K <- eigen(cor(Z))
> #Output skor dalam 2 digit
> print(K, digits = 2)
eigen() decomposition
$`values`
[1] 1.8 0.2

$vectors
     [,1]  [,2]
[1,] 0.71 -0.71
[2,] 0.71  0.71

> print(Z %*% K$vec, digits=2)
      [,1]   [,2]
[1,]  2.02 -0.290
[2,]  0.28  0.841
[3,]  1.34  0.028
[4,] -1.74  0.212
[5,] -0.80 -0.559
[6,] -1.09 -0.226
> # perform principal components analysis - Primpcomp
> pca <- princomp(Z, center = TRUE, cor=TRUE, scores=TRUE)
Warning message:
In princomp.default(Z, center = TRUE, cor = TRUE, scores = TRUE) :
 extra argument 'center' will be disregarded
> pca$scores
         Comp.1      Comp.2
[1,]  2.0148565  0.29277442
[2,]  0.2739993 -0.84028909
[3,]  1.3426687 -0.02604494
[4,] -1.7413036 -0.21239622
[5,] -0.7999818  0.55930866
[6,] -1.0902391  0.22664718
> #ex 2 -  The first five eigenvalues from the correlation matrix of g
olub
> eigen(cor(golub))$values[1:5]
[1] 25.4382629  2.0757158  1.2484411  1.0713373  0.7365232
> data <- golub; p <- ncol(data); n <- nrow(data) ; nboot<-1000
> eigenvalues <- array(dim=c(nboot,p))
> for (i in 1:nboot){dat.star <- data[sample(1:n,replace=TRUE),]
+ eigenvalues[i,] <- eigen(cor(dat.star))$values}
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

Analisis komponen utama sangat berguna untuk menemukan arah dalam data di mana nilai ekspresi gen bervariasi secara maksimal, lihat Jolliffe (2002) untuk perawatan lengkap dari analisis komponen utama. Saat ini rections dapat direpresentasikan dengan baik oleh dua komponen pertama yang membantu biplot secara simultan memvisualisasikan gen dan pasien. Analisis komponen utama dapat berguna dalam mengidentifikasi kelompok gen dalam ruang dimensi yang lebih rendah.

```
> for (j in 1:p) print(quantile(eigenvalues[,j],c(0.025,0.975)))

    2.5%    97.5%
24.81354 25.98087
    2.5%    97.5%
1.941043 2.252706
    2.5%    97.5%
1.145386 1.404069
     2.5%     97.5%
0.9946962 1.1478118
0.09351937 0.10695738
      2.5%      97.5%
0.08269396 0.09721315
> for (j in 1:5) cat(j,as.numeric(quantile(eigenvalues[,j],

+        c(0.025,0.975))),"\n" )
1 24.81354 25.98087
2 1.941043 2.252706
3 1.145386 1.404069
4 0.9946962 1.147812
5 0.6854926 0.7960533

> -eigen(cor(golub))$vec[,1:2]
             [,1]         [,2]
 [1,] 0.1715179  0.104190383
 [2,] 0.1690830 -0.036887326
 [3,] 0.1650130  0.069108652
 [4,] 0.1726783  0.100701410
 [5,] 0.1659430  0.170952365
 [6,] 0.1668802  0.028349089
 [7,] 0.1686381  0.032390622
   …
[34,] 0.1675337 -0.157687854
[35,] 0.1638383 -0.130649438
[36,] 0.1508646 -0.277921314
[37,] 0.1476138 -0.344828867
[38,] 0.1520466 -0.222765750
```

Fadhillah Putri Taha
H071171301

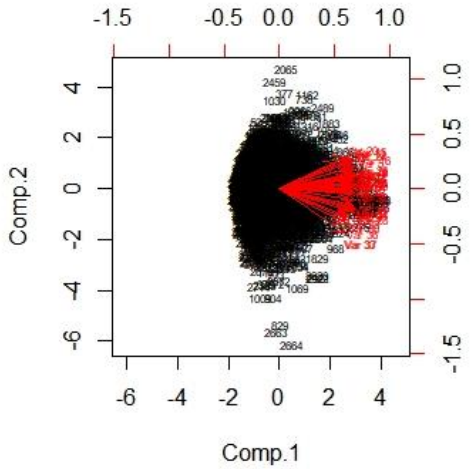
*Summary Cluster Analysis dan Trees*

```
> # The first and the last 10 gene names on the second component
> pca <- princomp(golub, center = TRUE, cor=TRUE, scores=TRUE)
Warning message:
In princomp.default(golub, center = TRUE, cor = TRUE, scores = TRUE) :
 extra argument 'center' will be disregarded
> o <- order(pca$scores[,2])
> golub.gnames[o[1:10],2]
 [1] "INTERLEUKIN-8 PRECURSOR"

 [2] "Interleukin 8 (IL8) gene"

 [3] "CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)"

 [4] "DF D component of complement (adipsin)"

 [5] "LGALS3 Lectin, galactoside-binding, soluble, 3 (galectin 3) (NOTE:
redefinition of symbol)"
 [6] "Azurocidin gene"

 [7] "GRO2 GRO2 oncogene"

 [8] "MACROPHAGE INFLAMMATORY PROTEIN 1-ALPHA PRECURSOR"

 [9] "LYZ Lysozyme"

[10] "CYSTATIN A"
> golub.gnames[o[3041:3051],2]
 [1] "LPAP gene"

 [2] "Cytoplasmic dynein light chain 1 (hdlc1) mRNA"

 [3] "RETINOBLASTOMA BINDING PROTEIN P48"

 [4] "Adenosine triphosphatase, calcium"

 [5] "C-myb gene extracted from Human (c-myb) gene, complete primary cds,
 and five complete alternatively spliced cds"
 [6] "IGB Immunoglobulin-associated beta (B29)"

 [7] "Terminal transferase mRNA"

 [8] "MB-1 gene"

 [9] "GB DEF = (lambda) DNA for immunoglobin light chain"

[10] "CD24 signal transducer mRNA and 3' region"

[11] "TCL1 gene (T cell leukemia) extracted from H.sapiens mRNA for Tcell
 leukemia/lymphoma 1"
```

Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

```
> #ex 3 - biplot which is based on a two-dimensional approximation
of the data

> biplot(princomp(data,cor=TRUE),pc.biplot=TRUE,cex=0.5,expand=0.8)
> #ex 4
> data(golub, package = "multtest")
> #Biplot of selected genes from the golub data
> factor <- factor(golub.cl)
> o1 <- grep("CD",golub.gnames[,2])
> o2 <- grep("Op",golub.gnames[,2])
> o3 <- grep("MCM",golub.gnames[,2])
> o <- c(o1,o2,o3)
> #use a two-sample t-test
> pt <- apply(golub, 1, function(x) t.test(x ~ gol.fac)$p.value)
> oo <- o[pt[o]<0.01]
> # to identify genes in directions of large variation
> # using scores on the first two principal components
> Z <- as.matrix(scale(golub, center = TRUE, scale = TRUE))
> K <- eigen(cor(Z))
> P <- Z %*% -K$vec[,1:2]
> leu <- data.frame(P[oo,], row.names= oo)
> attach(leu)
The following objects are masked from leu (pos = 3):

    X1, X2

The following objects are masked from leu (pos = 4):

    X1, X2

The following objects are masked from leu (pos = 5):

    X1, X2

The following objects are masked from leu (pos = 6):

    X1, X2

The following objects are masked from leu (pos = 7):

    X1, X2

The following objects are masked from leu (pos = 8):

    X1, X2
```



Fadhillah Putri Taha
H071171301

*Summary Cluster Analysis dan Trees*

```
> #Analysis Cluster Hierarchical

> cl <- hclust(dist(leu,method="euclidian"),method="single")
> plot(cl)
> #Gene Order
> a <- as.integer(rownames(leu)[cl$order])
> for (i in 1:length(a)) cat(a[i],golub.gnames[a[i],2],"\n")
1910 FCGR2B Fc fragment of IgG, low affinity IIb, receptor for (CD32)
2874 GB DEF = Fas (Apo-1, CD95)
2350 SLC6A8 gene (creatine transporter) extracted from Human Xq28 cosmid, cr
eatine transporter (SLC6A8) gene, and CDM gene, partial cds
1737 TFRC Transferrin receptor (p90, CD71)
1911 ME491  gene extracted from H.sapiens gene for Me491/CD63 antigen
2749 ITGAX Integrin, alpha X (antigen CD11C (p150), alpha polypeptide)
2761 CD36 CD36 antigen (collagen type I receptor, thrombospondin receptor)
792 ANPEP Alanyl (membrane) aminopeptidase (aminopeptidase N, aminopeptidase
 M, microsomal aminopeptidase, CD13)
2321 ICAM1 Intercellular adhesion molecule 1 (CD54), human rhinovirus recept
or
68 PRKCD Protein kinase C, delta
808 CD33 CD33 antigen (differentiation antigen)
2611 Cell division control related protein (hCDCrel-1) mRNA
450 Opioid-Binding Cell Adhesion Molecule
504 CD3Z CD3Z antigen, zeta polypeptide (TiT3 complex)
313 CD38 CD38 antigen (p45)
1756 CD3G CD3G antigen, gamma polypeptide (TiT3 complex)
893 CD72 CD72 antigen
182 NT5 5' nucleotidase (CD73)
2653 CD2 CD2 antigen (p50), sheep red blood cell receptor
2289 MCM3 Minichromosome maintenance deficient (S. cerevisiae) 3
2430 MCM3 Minichromosome maintenance deficient (S. cerevisiae) 3
316 P105MCM mRNA
1616 KAI1 Kangai 1 (suppression of tumorigenicity 6, prostate; CD82 antigen
(R2 leukocyte antigen, antigen detected by monoclonal and antibody IA4))
1101 CDC10 Cell division cycle 10 (homologous to CDC10 of S. cerevisiae
2673 CD19 gene
2459 CD24 signal transducer mRNA and 3' region
892 CD9 CD9 antigen
1882 CD22 CD22 antigen
885 CD53 CD53 antigen
849 Oncoprotein 18 (Op18) gene
2397 T-CELL ANTIGEN CD7 PRECURSOR
1754 Catalase (EC 1.11.1.6) 5'flank and exon 1 mapping to chromosome 11, ban
d p13 (and joined CDS)
1798 CD37 CD37 antigen
2233 GB DEF = CD36 gene exon 15
```