

Math7340 HW12

Chengbo Gu

Problem 1 (60 points) Analysis of the ALL data set

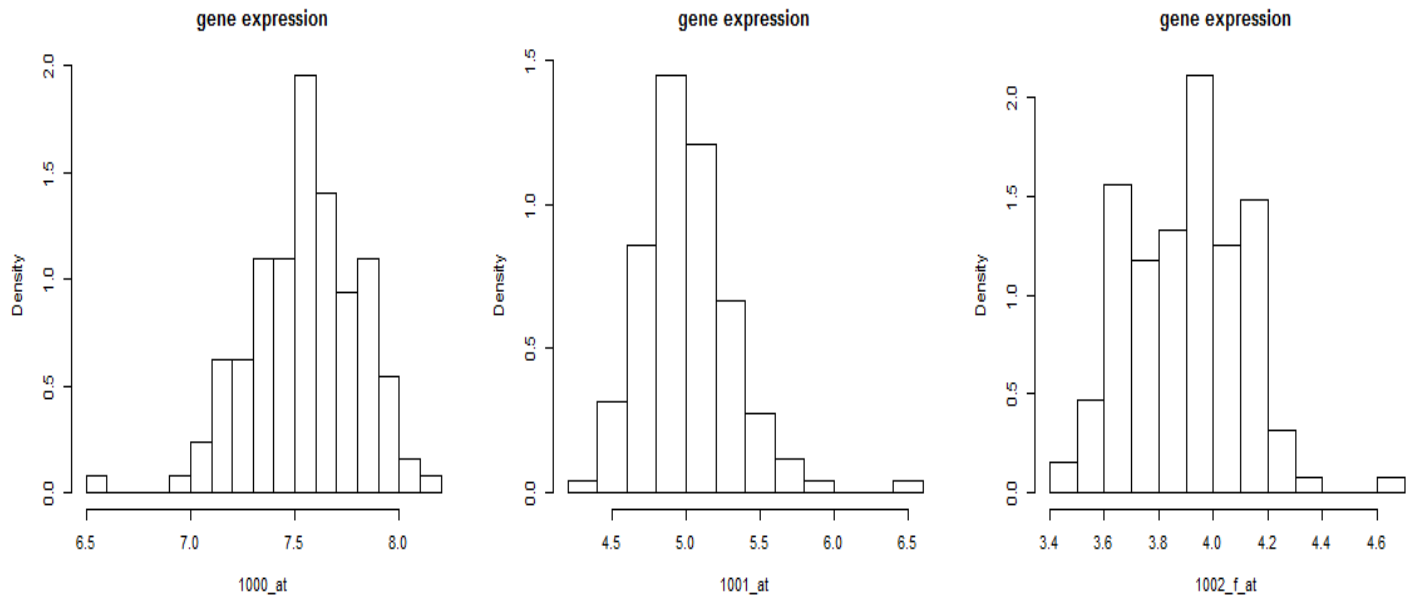
(a) Define an indicator variable *ALL.fac* such that *ALL.fac*=1 for T-cell patients and *ALL.fac*=2 for B-cell patients.

```
library(ALL)
```

```
data(ALL)
ALL.fac <- as.numeric(ALL$BT)
ALL.fac[ALL.fac <= 5] = 1
ALL.fac[ALL.fac > 5] = 2
```

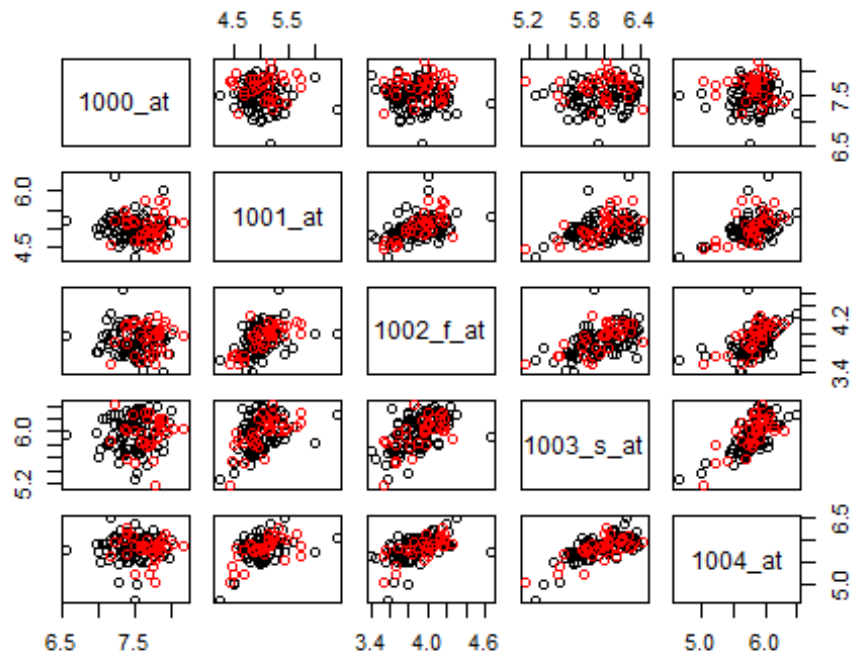
(b) Plot the histograms for the first three genes' expression values in one row.

```
par(mfrow=c(1,3))
for (i in 1:3) {
  hist(exprs(ALL)[i,], main="gene expression", nclass=15,
       freq=FALSE, xlab=rownames(exprs(ALL))[i])
}
```



(c) Plot the pairwise scatterplots for the first five genes.

```
firstFive <- t(exprs(ALL)[1:5,])
pairs(firstFive, col=ALL.fac)
```



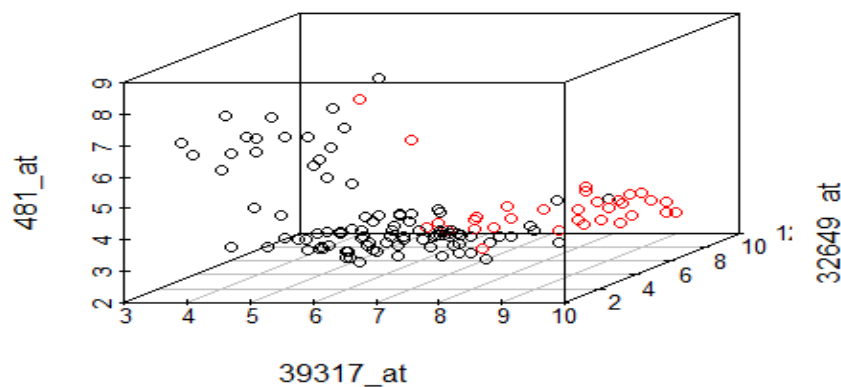
(d) Do a 3D scatterplot for the genes “39317_at”, “32649_at” and “481_at”, and color according to ALL.fac (give different colors for B-cell versus T-cell patients). Can the two patient groups be distinguished using these three genes?

```
rowname <- rownames(exprs(ALL))
index39317 <- grep("^39317_at$", rowname)
index32649 <- grep("^32649_at$", rowname)
index481 <- grep("^481_at$", rowname)

data <- t(exprs(ALL)[c(index39317, index32649, index481),])

require(scatterplot3d)

scatterplot3d(data, color=ALL.fac)
```



From the 3D plot, we conclude that two patient groups could be distinguished using these three genes.

(e) Do K-means clustering for K=2 and K=3 using the three genes in (d). Compare the resulting clusters with the two patient groups. Are the two groups discovered by the clustering analysis?

```
cl.2mean <- kmeans(data, centers=2, nstart = 10)
table(ALL.fac, cl.2mean$cluster)
```

```
##
## ALL.fac  1  2
##          1 21 74
##          2 31  2
```

```
cl.3mean <- kmeans(data, centers=3, nstart = 10)
table(ALL.fac, cl.3mean$cluster)
```

```
##
## ALL.fac  1  2  3
##          1 70  5 20
##          2  3 28  2
```

Yes, the two groups are discovered by K-means clustering when K = 3.

(f) Carry out the PCA on the ALL data set with scaled variables. What proportion of variance is explained by the first principal component? By the second principal component?

```
PCA <- prcomp(exprs(ALL), scale=TRUE)
summary(PCA)
```

```
## Importance of components:
##
##          PC1      PC2      PC3      PC4      PC5      PC6
## Standard deviation 10.9450 1.10132 0.93237 0.75341 0.62938 0.57412
## Proportion of Variance 0.9359 0.00948 0.00679 0.00443 0.00309 0.00258
## Cumulative Proportion 0.9359 0.94536 0.95215 0.95658 0.95968 0.96225
```

93.59% of variance is explained by the first principal component.

0.948% of variance is explained by the second principal component.

(g) Do a biplot of the first two principal components. Observe the pattern for the loadings. What info is the first principal component summarizing?

```
print(c(t(PCA$rotation[,1])), digits=3)
```

```
## [1] -0.0897 -0.0881 -0.0889 -0.0872 -0.0884 -0.0882 -0.0877 -0.0883
## [9] -0.0886 -0.0892 -0.0887 -0.0885 -0.0887 -0.0887 -0.0886 -0.0895
## [17] -0.0881 -0.0890 -0.0895 -0.0890 -0.0896 -0.0883 -0.0890 -0.0885
## [25] -0.0895 -0.0895 -0.0882 -0.0872 -0.0893 -0.0881 -0.0891 -0.0887
## [33] -0.0891 -0.0891 -0.0895 -0.0886 -0.0888 -0.0869 -0.0884 -0.0881
## [41] -0.0895 -0.0869 -0.0886 -0.0894 -0.0879 -0.0888 -0.0895 -0.0887
## [49] -0.0866 -0.0890 -0.0882 -0.0863 -0.0887 -0.0885 -0.0876 -0.0883
## [57] -0.0880 -0.0879 -0.0884 -0.0876 -0.0879 -0.0894 -0.0873 -0.0882
## [65] -0.0883 -0.0884 -0.0888 -0.0895 -0.0891 -0.0889 -0.0883 -0.0885
## [73] -0.0898 -0.0883 -0.0884 -0.0889 -0.0899 -0.0892 -0.0895 -0.0893
## [81] -0.0890 -0.0888 -0.0870 -0.0882 -0.0894 -0.0896 -0.0898 -0.0862
## [89] -0.0886 -0.0876 -0.0867 -0.0855 -0.0883 -0.0876 -0.0896 -0.0886
## [97] -0.0882 -0.0893 -0.0881 -0.0872 -0.0852 -0.0873 -0.0887 -0.0865
## [105] -0.0886 -0.0876 -0.0887 -0.0879 -0.0890 -0.0876 -0.0887 -0.0863
## [113] -0.0874 -0.0893 -0.0873 -0.0866 -0.0883 -0.0890 -0.0887 -0.0868
## [121] -0.0890 -0.0885 -0.0879 -0.0886 -0.0886 -0.0883 -0.0885 -0.0886
```



```
get(biggest, env = hgu95av2GENENAME)
```

```
## [1] "SNF related kinase"
```

```
get(biggest, env = hgu95av2CHR)
```

```
## [1] "3"
```

gene name and chromosome for the gene with smallest PC2 value

```
get(smallest, env = hgu95av2GENENAME)
```

```
## [1] "cytidine monophospho-N-acetylneuraminic acid hydroxylase, pseudogene"
```

```
get(smallest, env = hgu95av2CHR)
```

```
## [1] "6"
```

Problem 2 (40 points) Variables scaling and PCA in the iris data set

In this module and last module, we mentioned that the variables are often scaled before doing the PCA or the clustering analysis. By “scaling a variable”, we mean to apply a linear transformation to center the observations to have mean zero and standard deviation one. In last module, we also mentioned using the correlation-based dissimilarity measure versus using the Euclidean distance in clustering analysis. It turns out that the correlation-based dissimilarity measure is proportional to the squared Euclidean distance on the scaled variables. We check this on the iris data set. And we compare the PCA on scaled versus unscaled variables for the iris data set.

(a) Create a data set consisting of the first four numerical variables in the iris data set (That is, to drop the last variable Species which is categorical). Then make a scaled data set that centers each of the four variables (columns) to have mean zero and variance one.

```
data <- data.frame(iris[,1:4])
```

```
scaled.data <- scale(data)
```

```
colMeans(scaled.data)
```

```
## Sepal.Length Sepal.Width Petal.Length Petal.Width
```

```
## -4.480675e-16 2.035409e-16 -2.844947e-17 -3.714621e-17
```

```
apply(scaled.data, 2, sd)
```

```
## Sepal.Length Sepal.Width Petal.Length Petal.Width
```

```
## 1 1 1 1
```

The data is scaled now with mean 0 and sd 1.

(b) Calculate the correlations between the columns of the data sets using the cor() function. Show that these correlations are the same for scaled and the unscaled data sets.

```
cor(data)
```

```
## Sepal.Length Sepal.Width Petal.Length Petal.Width
```

```
## Sepal.Length 1.0000000 -0.1175698 0.8717538 0.8179411
```

```
## Sepal.Width -0.1175698 1.0000000 -0.4284401 -0.3661259
```

```
## Petal.Length 0.8717538 -0.4284401 1.0000000 0.9628654
```

```
## Petal.Width 0.8179411 -0.3661259 0.9628654 1.0000000
```

```
cor(scaled.data)
```

```
##           Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length      1.0000000  -0.1175698   0.8717538   0.8179411
## Sepal.Width       -0.1175698   1.0000000  -0.4284401  -0.3661259
## Petal.Length      0.8717538  -0.4284401   1.0000000   0.9628654
## Petal.Width       0.8179411  -0.3661259   0.9628654   1.0000000
```

These correlations are the same for scaled and the unscaled data sets.

(c) Calculate the Euclidean distances between the columns of the scaled data set using `dist()` function. Show that the squares of these Euclidean distances are proportional to the $(1 - \text{correlation})$ s. What is the value of the proportional factor here?

```
dist(t(scaled.data), method="eucl")

##           Sepal.Length Sepal.Width Petal.Length
## Sepal.Width      18.249268
## Petal.Length      6.182020  20.631896
## Petal.Width      7.365701  20.176856   3.326575

d <- c(dist(t(scaled.data), method="eucl")^2)
corr <- cor(scaled.data)[lower.tri(diag(4))]
d/(1-corr)
```

```
## [1] 298 298 298 298 298 298
```

The proportional factor here is 298.

(d) Show the outputs for doing PCA on the scaled data set and on the unscaled data set. (Apply PCA on the two data sets with option `"scale=FALSE"`. Do NOT use option `"scale=TRUE"`, which will scale data no matter which data set you are using.) Are they the same?

```
unscaled.PCA <- prcomp(data, scale=FALSE)
scaled.PCA <- prcomp(scaled.data, scale=FALSE)

summary(scaled.PCA)

## Importance of components:
##           PC1      PC2      PC3      PC4
## Standard deviation  1.7084 0.9560 0.38309 0.14393
## Proportion of Variance 0.7296 0.2285 0.03669 0.00518
## Cumulative Proportion 0.7296 0.9581 0.99482 1.00000

summary(unscaled.PCA)

## Importance of components:
##           PC1      PC2      PC3      PC4
## Standard deviation  2.0563 0.49262 0.2797 0.15439
## Proportion of Variance 0.9246 0.05307 0.0171 0.00521
## Cumulative Proportion 0.9246 0.97769 0.9948 1.00000
```

They are not the same.

(e) What proportions of variance are explained by the first two principle components in the scaled PCA and in the unscaled PCA?

scaled:

PC1: 72.96%

PC2: 22.85%

PC1+PC2: 95.81%

unscaled:

PC1: 92.46%

PC2: 5.307%

PC1+PC2: 97.769%

(f) Find a 90% confidence interval on the proportion of variance explained by the second principal component, in the scaled PCA.

```
#scaled.PCA$sdev^2/sum(scaled.PCA$sdev^2)
p <- ncol(scaled.data)
n <- nrow(scaled.data)
nboot<-1000
sdevs <- array(dim=c(nboot,p)) #matrix to save resampled sdev for p PCs
for (i in 1:nboot) {
  dat.star <- scaled.data[sample(1:n,replace=TRUE),] #resample rows
  sdevs[i,] <- prcomp(dat.star)$sdev #sdev for PCA on resampled data
}
as.numeric(quantile(sdevs[,2], c(0.05,0.95)))/(sum(scaled.PCA$sdev^2))

## [1] 0.2102140 0.2603928
```