Lecture #3

Data visualizations, outliers, and missing data

R Code

```
# scatter plot matrix
dat <- read.table("C:\\gecolon.dat",header=T)
dimnames(dat)[[1]] <- as.character(dat[,1])
dat <- dat[,-1];        dat <- as.data.frame(dat);

# other data sets in R to use
library(Biobase);       library(annotate);      library(golubEsets);
data(golubTrain);        data(golubTest);       data(geneData);
dat <- geneData or dat <- exprs(golubTrain) or dat <- exprs(golubTest)

# box plots
boxplot(dat,cex=0.45,col='red',main="Box plots-Tumor data")

# random selection of 5 samples
rand.sams <- sample(names(dat),5,replace=F)
# plot trellis
pairs(dat[,rand.sams])

# Pearson's correlation matrix
dat.cor <- cor(dat)
image(dat.cor,axes=F)
axis(2,at=seq(0,1,length=ncol(dat.cor)),label=dimnames(dat.cor)[[2]])
axis(3,at=seq(0,1,length=ncol(dat.cor)),label=dimnames(dat.cor)[[2]])

# random sample of 5 genes
rand.genes <- sample(dimnames(dat)[[1]],5,replace=F)

# profile plot
plot(c(1,ncol(dat)),range(dat[rand.genes,]),type='n',main="Profile plot of 5 random
     genes",xlab="Samples",ylab="Expression")
for(i in 1:length(rand.genes)) {
     dat.y <- as.numeric(dat[rand.genes[i],])
     lines(c(1:ncol(dat)),dat.y,col=i)
}
```

R Code

```
# load the yeast cell cycle data set
dat <- read.table("C:\\spellman.txt",header=T)
dimnames(dat)[[1]] <- as.character(dat[,1])
dat <- dat[,-1]
dat <- dat[,23:46]
dat[is.na(dat)] <- 0

# pca biplot
biplot(prcomp(t(dat[500:550,])),cex=0.6)

# k-means cluster profiles
dd <- dat[names(f.p)[f.p<0.001],]
d.k <- kmeans(dd,9)
par(mfrow=c(3,3))
for(i in 1:9) {
     tmp <- scale(dd[d.k$cluster==i,])
     matplot(c(1:ncol(dat)),t(tmp),type='l',col=i,xlab='Time',ylab='Expression')
}
# cv vs. mean plot
dat.mean <- apply(dat,1,mean)                    # calculate mean for each gene
dat.sd <- sqrt(apply(dat,1,var))                 # calculate st.deviation for each gene
dat.cv <- dat.sd/dat.mean                        #calculate cv

plot(dat.mean,dat.cv,main="Sample CV vs. Mean",xlab="Mean",ylab="CV",col='blue',cex=1.5)

# 2D sample pca plot
dat.pca <- prcomp(t(dat))
dat.loads <- dat.pca$x[,1:2]
plot(dat.loads[,1],dat.loads[,2],main="Sample PCA plot",xlab="p1",ylab="p2",col='red',cex=1.5,pch=16)
```

R Code

```
# k-means clustering for missing value imputation
dat <- dat[2:30,]                                          # only use 29 genes for example
cl <- kmeans(dat[,-1],centers=5, iter.max=20)              # cluster into 5 groups
                                                 # we pretend to be missing a value at sample#1 gene #2
groups <- cl$cluster                             # get cluster membership for each gene
groups                                           # look at groups to see where gene 2 is
group.2 <- groups==2                             # since gene 2 is in group 2, get all other members
genes.cluster <- dimnames(dat)[[1]][group.2]
genes.cluster                                    # look at all other genes in cluster #2

gene.dist <- dist(dat[genes.cluster,-1],method="euclidean") # get distances from genes in cluster 2 to
                                                 # gene #2
gene.dist <- as.matrix(gene.dist)
gene.dist <- gene.dist[2:5,1]
gene.weight <- as.numeric(gene.dist/sum(gene.dist))        # get weights for each gene

weight.mean <-  weighted.mean(dat[genes.cluster[-1],1], gene.weight)    # calculate weighted mean for
                                                           # gene #2


# perspective plot
data(volcano)           # load volcano data set
persp(volcano, theta=45, phi=30, col="red")

# MvA plot
library(sma)
data(MouseArray)
mouse.lratio <- stat.ma(mouse.data, mouse.setup)
plot.mva(mouse.data, mouse.setup, norm="l", 2, extra.type="pci", plot.type="n",main="MvA plot")
```

R Code

```
# calculate mean for some genes, with respect to class
library(multtest)
data(golub)
dat <- as.data.frame(golub)
ann <- golub.cl
dat.aml <- apply(dat[,ann==1],1,mean)
dat.all <- apply(dat[,ann==0],1,mean)
tab <- data.frame(rbind(dat.aml[1:20],dat.all[1:20]))
dimnames(tab)[[1]] <- c("AML","ALL")
names(tab) <- dimnames(dat)[[1]][1:20]
mp <- barplot(tab)
tot <- colMeans(tab)
text(mp, tot + 3, format(tot), xpd = TRUE, col = "blue")
barplot(as.matrix(tab),beside=T,col=c("red","yellow"),legend=rownames(as.matrix(tab)),ylim=c(-
    5,5),ylab="Expression")
title(main = "Mean Expression Levels of first 20 genes")

# cluster tree
dat <- t(dat)                                    #transpose dat
dat.dist <- dist(dat,method="euclidean")         # calculate distance
dat.clust <- hclust(dat.dist,method="single")    # calculate clusters
plot(dat.clust,labels=names(dat),cex=0.75)                 # plot cluster tree
```