**HW 3 type 3 – Q and A: Clustering of biological data and phylogenetic trees - 2013**

**Introduction**

In this homework you will gain some skills on clustering biological data using various algorithms and how to represent data graphically. Clustering is a natural concept to humans as our brains are inherently designed to classify and group data. Sometimes we fall into prejudices and pre-mature conclusions that in bioinformatics and data-mining terms could be translated to concepts such as mis-classification or clustering errors. Please answer below questions that will guide you through path of discovering different clustering techniques under biological context.

**Instructions:** Complete the questions below. You can answer questions below or create a new report. In the report please do not forget to include figures and tables and your source code (that could be put in the appendix). Read provided references that will give you a better understanding of data and methods allowing you to better develop discussions

**Due date:** January 8th, 2014

**Useful links:**

Uniprot database for protein sequence retrieval: <http://www.uniprot.org/uniprot/>

**Questions to be answered**

**Phylogenetic trees**

**This part of homework you would develop your own R-code to answer the questions below that is very similar to the one shown during the lecture. Consult lecture notes. Provide your R-code in report**

1. Calculate the genetic distances between the following gag HIV structural proteins of different types and different isolates: a) Unitprot ID: **P04590** - Human immunodeficiency virus type 2 subtype A (isolate ROD) (HIV-2); b) Unitprot ID: **P24736** - Human immunodeficiency virus type 1 group M subtype A (isolate U455) (HIV-1); c) Unitprot ID: **P15832** - Human immunodeficiency virus type 2 subtype B (isolate D205) (HIV-2); d) Unitprot ID: **P04591** - Human immunodeficiency virus type 1 group M subtype B (isolate HXB2) (HIV-1); e) Unitprot ID: **Q74230** - Human immunodeficiency virus type 2 subtype B (isolate EHO) (HIV-2); f) Unitprot ID: **Q77372** - Human immunodeficiency virus type 1 group O (isolate ANT70) (HIV-1)
	1. Which are the most closely related proteins (smallest distance), and which are the least(larger distance) closely related, based on calculated genetic distances?
	2. From the 6 *gag* proteins, can one draw a conclusion that HIV-1 and HIV-2 type viruses represent 2 distinct groups (and thus cluster into 2 distinct groups) or there is an overlap existing between groups?
2. Build an **unrooted** phylogenetic tree of the six HIV gag proteins mentioned in previous question using the neighbour-joining algorithm (NJ). Which are the most closely related proteins, based on the tree?
3. Build an **rooted** phylogenetic tree of the six HIV gag proteins using the neighbour-joining algorithm (NJ). Do you see two clusters of two types of *gag* HIV proteins (i.e. do you see that these two types proteins are closer together to the proteins of the same type or not)? Are those two clusters relatively “pure” (i.e. if you were to create two clusters/groups, do you see some “contamination” in those groups from other classes?). Describe the tree.
4. Based on the rooted tree results, how similar and different are these 6 HIV proteins amongst themselves? Which proteins show the closes similarity between them (provide Uniport IDs or protein names)?

**Unsupervised clustering**

In this section you will be clustering samples coming from individuals with different type of cancer. **For your convenience, the code is provided below due to complexity of the dataset**. You will be provided with expression data (microarrays) and sample labels (type of cancer an individual has). The dataset is based on the publication by Ross DT [1] and data from [Stanford NCI60 Cancer Microarray Project](http://genome-www.stanford.edu/nci60/). The dataset is supplied as **nci60.data.txt** tab-delimited text file [2]. The dataset is composed of 64 samples and 6840 genes. Please note that labels for samples are column names while rows represent genes. Your job will be clustering this data in unsupervised way as if you did not know the sample labels. **Unsupervised clustering** is the one that uses original data structure and predicts sample class based on some algorithm without any introduction of previous knowledge(i.e. sample labels a not known to algorithm)[3]. Assuming that clinicians had not done any error, the sample labels (i.e. cancer types) are golden reference standard and are considered as ultimate truth. This will be important when we will compare results of clustering via k-means against true sample labels.

1. **K-means**
2. Describe how k-means clustering algorithm works and why fluctuation in clustering results could be expected even if k-means is run on identical data again?
3. What applications is it used for?
4. What are its advantages and disadvantages?

Read the input data using R into a matrix and run k-means clustering function with k=14 to match number of classes. Observe the results and answer the following questions. Do not forget to point R to read files from the correct directory using *setwd()* function.

1. Determine if k=14 is an optimal number of clusters? What does plot Within Group Sum of Squares (Errors) vs number of clusters suggests? How does the results from the previous plot can be linked to data complexity? Remember, the lower the Within Group Sum of Squares the better defined clusters are.

uA\_data <- read.table(file="nci60.data.txt", header=T, check.names=F)

uA\_data = t(uA\_data) #transpose data for k-means processing

uA\_data\_kmeans\_14 <- kmeans(uA\_data , centers=14, iter.max=100)

uA\_data\_kmeans\_14 #the k-means object

wss=rep(0,24); k=rep(0,24)

for(i in 1:24){

 cat("Testing with k=",i+1,"\n"); flush.console();

 k[i]=i+1

 wss[i] <- kmeans(uA\_data , centers=i+1, iter.max=100)$tot.withinss

}

plot(k,wss, ylab="Within Group Sum of Squares (Errors)"); lines(k,wss)

axis(1,at=1:24)

1. Was k-means was able to identify 14 clusters/sample types in the data? Examine output of the k-means object, namely its *clustering vector*. Show the output below (copy and paste)

clusters\_kmeans <- uA\_data\_kmeans\_14$cluster

1. As seen from previous output, it is hard to deduce the identity of the cluster numbers as k-means algorithm that did not know a priori the sample labels (unsupervised clustering). The top labels refer to the true sample labels as supplied to us (i.e. cancer types). One way to solve this problem is to hope for good separation of samples and good accuracy in classification of the data. For each cluster number (1-14) we will try to deduce its identity by looking at the majority cluster number assigned to a given cancer type (i.e. class label). E.g. we will extract all samples of LEUKEMIA cancer type and look at their cluster number assignments and select the most frequent one. **After execution of the code below, was k-means able unambiguously assign all 64 samples to 14 clusters? Based on majority class, were some classes shared/duplicated amongst cancer types? Which cluster numbers were duplicated? Which cancer type is particularly was hard to cluster (i.e. no majority class was found)?**

cancers <- unique(names(clusters\_kmeans))

table\_maj\_class <- matrix(0,nrow=14,dim=list(cancers,"maj.class"))

for(cancer in cancers){

 assignments <- clusters\_kmeans[which(names(clusters\_kmeans)==cancer)]

 #cat(names(assignments), "\n", sep="\t"); #cat(assignments, "\n", sep="\t")

table(assignments)

 majority\_class <- names(table(assignments)[table(assignments)==max(table(assignments))])

 #cat("Majority class =", majority\_class, "\n")

 if (length(majority\_class) == 1){table\_maj\_class[cancer,]=as.numeric(majority\_class)

 }else{table\_maj\_class[cancer,]=NA}

}

table\_maj\_class

1. Given the true labels of the columns (ie true sample labels), what are approximate misclassification/error rates only for LEUKEMIA and MELANOMA samples?

select\_cancers <- clusters\_kmeans[which(names(clusters\_kmeans)=="LEUKEMIA")]

cluster\_num = table\_maj\_class["LEUKEMIA",]

num\_samples\_OK\_classif = select\_cancers[which(select\_cancers==cluster\_num)]

num\_samples\_FAIL\_classif = length(select\_cancers) - length(num\_samples\_OK\_classif)

cat("Misclassification error (decimal)",num\_samples\_FAIL\_classif/length(select\_cancers) ,"\n")

select\_cancers <- clusters\_kmeans[which(names(clusters\_kmeans)=="MELANOMA")]

cluster\_num = table\_maj\_class["MELANOMA",]

num\_samples\_OK\_classif = select\_cancers[which(select\_cancers==cluster\_num)]

num\_samples\_FAIL\_classif = length(select\_cancers) - length(num\_samples\_OK\_classif)

cat("Misclassification error (decimal)",num\_samples\_FAIL\_classif/length(select\_cancers) ,"\n")

1. **Principal component analysis (PCA)**
2. What is PCA and what is the main objective of PCA? Why PCA is applied to complex datasets?
3. Explain how optimal number of components is selected? What is the main criterion of determining the number of selected components for future analysis? What is the ideal case?
4. What are residuals represent in PCA analysis (term linked to regression also)? Why residuals could be useful for future analysis? (please consult reference on PCA [4])
5. Run PCA using R function *prcomp()* with parameter *retx=TRUE* on the untransformed initial input data (64 samples and 6840 genes present in **nci60.data.txt**). Visualize data projected on first 2 principal components (see option *retx*). **Include the figure of this plot**. Can you easily visually identify 14 clusters or it is not clear from the plot? What do you think is complex about clustering this dataset? How much of cumulative proportion of data **variance** do these 2 PCA components capture (i.e. provide exact value)? How this complexity of this dataset is linked to cumulative proportion of variance explained if looking only at 1st two PCA principal components (PCs)?

summary(PCA)

plot(PCA$rotation[,1], PCA$rotation[,2])

1. Does clustering performance improved (i.e. lower Within Group Sum of Squares error) than before (no PCA transformation)? (compare results with section A)

uA\_data <- read.table(file="nci60.data.txt", header=T, check.names=F)

uA\_data = t(uA\_data)

PCA = prcomp(uA\_data, retx=T )

PCA\_uA\_data\_kmeans\_14 <- kmeans(PCA$rotation , centers=14, iter.max=100, nstart=1)

wss=rep(0,24); k=rep(0,24)

for(i in 1:24){

 cat("Testing with k=",i+1,"\n"); flush.console();

 k[i]=i+1

 wss[i] <- kmeans(PCA$rotation , centers=i+1, iter.max=100)$tot.withinss

}

plot(k,wss, ylab="Within Group Sum of Squares (Errors)"); lines(k,wss)

axis(1,at=1:24)

**References:**

[1] Ross DT et.al. **Systematic variation in gene expression patterns in human cancer cell lines**. Nat Genet. 2000 Mar;24(3):227-35. (posted on the website, password protected)

[2] Reference homework on clustering <http://www.stat.cmu.edu/~cshalizi/350/hw/04/hw-04.pdf>

[3] Nizar Grira **Unsupervised and Semi-supervised Clustering: a Brief Survey** <http://cedric.cnam.fr/~crucianm/src/BriefSurveyClustering.pdf>

[4] SVANTE WOLD **Principal Component Analysis** *Chemometrics and Intelligent Laboratory Systems*, 2 (1987) 37