Exploratory data analysis - practical

Wessel N. van Wieringen

2019-06-24

1. Preliminaries

First load the required libraries and the dataset used in the clustering.

Load the necessary libraries.

```
library(marray)
library(affy)
library(ConsensusClusterPlus)
library(CLL)
```

Load the data into memory.

```
data(sCLLex)
```

The data comprises 22 CLL (Chronic Lymphocytic Leukemia) samples. Each sample has been profiled transcriptomically using an Affymetrix array, which yielded expression levels of 12,625 genes.

Filter the genes on variance.

```
Y <- exprs(sCLLex)[which(apply(exprs(sCLLex), 1, sd) > 1.75), ]
```

Warning: This type of filtering is a common but not necessary good practice! In particular, the filtering criterion is completely arbitrary. You may explore other criteria.

2. Hierarchical clustering

Let’s cluster the samples hierarchically. First, a distance matrix is calculated. This matrix containes the similarities among the samples. The next line of code then forms the dendrogram that captures the similarities among clusters of samples. It does so sequentially. Starting from the situation where each sample forms a cluster, the cluster pair that is most similar are merged to form a new cluster. This process is repeated until a single cluster remains. The last line of code plots the dendrogram, thereby displaying when two clusters have been merged.

```
distMat <- dist(t(Y),      method="manhattan")
dendro  <- hclust(distMat, method="complete")
plot(dendro, labels=pData(sCLLex)[,2])
```

In the above the similarities among samples have been calculcated by the Manhattan distance, while clusters have been merged using complete-linkage. As the goal of exploratory data analysis is high-light salient features in the data, any choice is as good as another.

Question: Try other linkages and distances. Type `?dist` and `?hclust` to see which have been implemented. Do results, i.e. the dengrogram, for other choices differ dramatically?

The dendrogram is often plotted in combination with a heatmap, a visual representation of the data, to provide some insight into the structure of the dendrogram in terms of the data. In the code below the color scheme of the heatmap is set prior to its plotting alongside the dendrogram.

```
pal <- maPalette(low="blue", high="yellow", mid="grey", k=25)
heatmap(Y, Colv=as.dendrogram(dendro), col=pal, labCol=pData(sCLLex)[,2])
```

Question: How many clusters would you conclude to be present among the samples from the dendrogram? With your chosen number of cluster, try – by eyeballing – to contrast in the row-wise direction the data between any two clusters. Could you pinpoint a gene or a group of genes that clearly contribute to the seperation at the sample level?

Once you have decided on the number of clusters present in the data you can find the samples that constitute them. Hereto cut dendrogram into (say) two clusters.

```
classes <- cutree(dendro, k=2)
```

The clusters are arbitrary labelled 1 and 2 (and subsequent integers for larger number of clusters).

3. Principal component analysis

Principal component analysis is a technique to summarize the data. It represents the 22 samples not by the information of the 12,625 genes, but by a handful of (linear) combinations of the genes' information. This allows for 2-dim visualization of the data to identify e.g. outlying samples or groups of samples.

Perform principal component analysis

```
svdRes <- svd(Y)
```

Generate principal component plot with the first two principle components. Other components may be selection by replacing `c(1,2)` by an other vector of two indices.

```
plot(svdRes$v[, c(1,2)], xlab="1st PC", ylab="2nd PC",  main="Principal comp
onent plot")
```

Study the plot for salient features.

Use PCA to confirms the number of discovered clusters (say, 2) found by hierarchical clustering.

```
# Plot different labels for each cluster
classes <- cutree(dendro, k=2)
points(svdRes$v[classes==1, c(1,2)], pch=20, col="red")
points(svdRes$v[classes==2, c(1,2)], pch=20, col="blue")
```

Question: Does the principal component-plot confirm your clustering?

Question: Do both principal components contribute to the cluster separation? Try with three principal components and make pairwise principal component plots.

4. K-means

K-means is another cluster technique. In contrast to hierarchical cluster the user specificies the number of clusters prior to the analysis. For example, one may be convinced there are K=3 clusters. Starting from user-specified or randomly chosen cluster representatives (i.e. the means from the nomen) the method assign samples to clusters (on the basis of their distance to the cluster representatives). Subsequently, representatives are updated on the basis of the samples that are assigned to then, after which samples are re-assigned. This process is iterated until convergence.

To perform K-means run:

```
kmeansRes <- kmeans(t(Y), centers=2)
```

The argument `centers` is the K from the method.

Link the found clusters to clinical data.

```
table(pData(sCLLex)[,2], kmeansRes$cluster)
```

Question: Perform do clinical data endow the clusters with a clear interpretation?

It is often good to perform K-means with several different initial representatives/means. This will eleminate a lucky, random clustering.

```
kmeansRes10 <- kmeans(t(Y), centers=2, nstart=10)
```

Does the repeated K-means clustering result differ from the single K-means clustering?

```
table(pData(sCLLex)[,2], kmeansRes10$cluster)
```

Question: Try with more centers, i.e. a different K.

5. Consensus clustering

In principle, it is not possible to assess the number of clusters from the data used in clustering. The validation of the number of clusters thus needs to be done on novel data. But … it may be sensible to choose a number of clusters that is robust (to perturbations). This will increase the probability that the clustering will reproduce on the novel data. This perturbation can be introduced in the data at hand in various ways (e.g. resampling, adding noise). The former – resampling – is exploited by consensus clustering, which allows the resampling of both samples and features. Samples that cluster together over many resamples form a stable cluster. This stability gives guidance in the number of clusters to be chosen in order to maximize chances of reproducing in novel data: a more stable clustering is to be preferred.

Perform consensus clustering.

```
ccRes <- ConsensusClusterPlus(dist(t(Y), method="manhattan"),
                              maxK=5, reps=100, pItem=0.8,
                              pFeature=1, clusterAlg="hc",
                              innerLinkage="complete")
```

The code above uses hierarchical clustering (`clusterAlg="hc"`) with the manhattan-distance (`method="manhattan"`) and complete-linkage (`innerLinkage="complete"`). From the dendrogram maximally five clusters (`maxK=5`), i.e. two to five clusters, are chosen by means of the ruler. The stability is assessed in 100 resamples (`reps=100`) drawing 80% of samples (`pItem=0.8`) each time, but does not subsample the features (`pFeature=1`).

Make heatmap of pairwise stability frequency.

```
pal <- maPalette(low="white", high="red", k=25)
heatmap(ccRes[[2]]$consensusMatrix, scale="none",
        labCol=pData(sCLLex)[,2], labRow=pData(sCLLex)[,2], col=pal)
```

Question: Check for other number of clusters. Which do you prefer?

6. Breast cancer data

Try the famous breast cancer example yourself:

```
library(hybridHclust)
library(marray)
data(sorlie)
data(sorlielabels)
```

7. AML/ALL data

Or, the AML/ALL data set used in the lecture.

```
library(multtest)
data(golub)
```