Supplementary examples

Download the file mix.out from http://maths.nuigalway.ie/~cathal/mix.out.gz

This file contains RNA-Seq data from the GTEx consortium. Samples are from three different tissues: brain, pancreas and skeletal muscle. Some of the brain samples are from the Cerebellum, the remainder are from the Hippocampus. The first 9 columns of data correspond to three different samples from each of these tissues (all of the pure brain samples are from the Hippocampus). The remaining data columns correspond to simulated mixtures of the same tissues with varying mixture proportions. Importantly, each mixture is constructed from a different set of pure tissue samples and all are distinct from the 9 pure samples.

The brain samples in the mixtures come from two brain regions – the cerebellum and the hippocampus.

Download the file brainsamples from http://maths.nuigalway.ie/~cathal/brainsamples.gz . This contains the pure brain samples from the mixtures and will be used later to compare to cell type specific differential expression results derived from the mixed samples.

```
library(CellMix)

# Save the files in your current working directory, uncompress them (e.g. using gunzip) and load into R.
data <- as.matrix(read.table("mix.out", row.names=1))
pure <- data[,1:9]
mixed <- data[,10:21]

# True mixture proportions
true <-
matrix(c(0.1,0.1,0.1,0.1,0.2,0.4,0.8,0.05,0.05,0.15,0.1,0.75,0.1,0.3,0.5,0.8,0.1,0.1,0.1,0.2,0.45,0.05,0.85,0.1,0.8,0.6,0.4,0.1,0.7,0.5,0.1,0.75,0.5,0.8,0.05,0.15),byrow=T, nrow=3)
rownames(true) <- c("brain","pancreas","muscle")

# Assigning markers the hard way…
brain_min <- apply(pure[,1:3],1,min)
pancreas_min <- apply(pure[,4:6],1,min)
muscle_min <- apply(pure[,7:9],1,min)
nonbrain_max <- apply(pure[,4:9],1,max)
nonpancreas_max <- apply(pure[,c(1:3,7:9)],1,max)
nonmuscle_max <- apply(pure[,1:6],1,max)
u <- 20
l <- 0.1
brain_mark <- rownames(pure[brain_min > u & nonbrain_max < l,])
pancreas_mark <- rownames(pure[pancreas_min > u & nonpancreas_max < l,])
muscle_mark <- rownames(pure[muscle_min > u & nonmuscle_max < l,])
```

```
markers <- MarkerList(list(brain = brain_mark, pancreas = pancreas_mark,
muscle = muscle_mark))
barplot(markers)

barplot(markers,pure)

# semi-supervised NMF with Frobenius norm
res <- ged(mixed, markers,me="ssFrobenius")
profplot(res, true)

# semi-supervised NMF with Kullback-Leibler divergence
res = ged(mixed, markers,method="ssK")
profplot(res, true)

# deconf unsupervised method by Repsilber et al. Note that it still requires the
markers for post-hoc assignment to cell types
res = ged(mixed, markers, method="deconf")
profplot(res, true)

# Digital sorting algorithm of Zhong et al.
res = ged(mixed, markers,method="DSA")
profplot(res, true)

# Accuracy of the basis estimation
brain <- read.table("brainsamples", row.names=1)
brainregion <- factor(c(rep("Cerebellum",6),rep("Hippocampus",6)))
brainp <- apply(brain, 1, function(brain) {   t.test(brain~brainregion)$p.value})
brainfdr <- p.adjust(brainp,me="fdr")
brainmean = apply(brain,1,mean)
df = data.frame(x = log(brainmean), y = log(basis(res)[,1]))
# you need ggplot2 for the next plot. Alternatively, you could make a similar plot
using R's plot function
require(ggplot2)
ggplot(aes(x=x,y=y),data=df) +
geom_point(alpha=0.1)+geom_abline(intercept=0,slope=1,col="red") +
xlab("Mean of brain samples") + ylab("Brain component of basis matrix")
```

Cell type specific differential expression analysis

```
# using csSAM from Shen-Orr et al. through CellMix (taking proportion estimates
from DSA). This could take a couple of minutes – time enough to read the help
page and really understand what csSAM is doing
csres = ged(mixed, coef(res),method="csSAM",data=brainregion)

# view the most significant genes (and associated FDRs) for each cell type
csTopTable(csres,n=20)
csplot(csres)

# Install CellCODE from http://www.pitt.edu/~mchikina/CellCODE/
```

```
# construct a table of the mean expression levels of each gene in the pure tissues (brain, pancreas, muscle)
library(CellCode)
puremean <-
cbind(apply(pure[,1:3],1,mean),apply(pure[,4:6],1,mean),apply(pure[,7:9],1,mean))
colnames(puremean) <- c("brain","pancreas","muscle")

# CellCODE - identify 'tag genes'
cctag <- tagData(puremean, max=15)

# CellCODE - identify 'surrogate proportion variables'
SPVs <- getAllSPVs(mixed,brainregion,cctag,plot=T)

# CellCODE – fit linear models for the variable of interest, with SPVs as covariates
ttcellcode <- lm.coef(mixed,model.matrix(~1+brainregion+ SPVs))

# p values for differentially expressed genes specific to brain
cellcodep = ttcellcode$pval[,2]

# Assign DE genes to cell types with CellCODE. This computes the F statistic described in the CellCODE paper (Chikina et al. 2015) to assign DE genes to cell types
resf <- cellPopF(mixed[cellcodep < 0.05,],brainregion, SPVs)

# How well did we do?
# compare the top 1000 'DE' genes for each of three methods (csSAM, CellCODE and uncorrected) to the 1000 most 'significantly' differentially expressed genes from the direct comparison of the unmixed brain samples.
brainp = sort(brainp)[1:1000]

# csSAM
cstt = csTopTable(csres,n=1000)
csp = cstt$brain
length(intersect(names(csp),names(brainp)))

# CellCODE
cellcodep = sort(cellcodep)[1:1000]
length(intersect(names(cellcodep),names(brainp)))

# Uncorrected
mixedp <- apply(mixed, 1, function(mixed) {
t.test(mixed~brainregion)$p.value})
mixedp = sort(mixedp)[1:1000]
length(intersect(names(mixedp),names(brainp)))
```

Uncorrected wins!! So why have we bothered?? What's going on? Discuss ☺