PBMC Deconvolution

Greg Hunt

June 6, 2018

There is a good reference for PBMCs from a 2015 paper by Newman et al. They call the reference LM22. You can access this data as part of the dtangle.data package available through our webpage: http://dtangle.github.io

First we load the packages

```
library('dtangle')
library('dtangle.data')
library('limma')
```

then we can load the Newman PBMC data set

```
dset = newman_pbmc
```

then load gene expressions (data) and mixture proportions (mix)

```
data = dset$data$log
data[1:5,1:5]
```

```
##                A1CF      A2M   A4GALT    A4GNT     AAAS
## GSM1587800 4.445675 4.333151 4.297844 4.293793 4.336305
## GSM1587801 4.361775 4.306877 4.263006 4.467572 4.380213
## GSM1587802 4.543298 4.328266 4.302650 4.809243 4.626695
## GSM1587803 4.568712 4.307912 4.302547 4.925474 4.329159
## GSM1587804 4.492878 4.274979 4.375282 4.738910 4.306863
```

```
mix = dset$annotation$mixture
mix[1:5,1:5]
```

```
##            B cells memory B cells naïve Dendritic cells activated
## GSM1587800         0.0296        0.1236                          0
## GSM1587801         0.0183        0.0333                          0
## GSM1587802         0.0407        0.1509                          0
## GSM1587803         0.0295        0.1677                          0
## GSM1587804         0.0469        0.0840                          0
##            Dendritic cells resting Eosinophils
## GSM1587800                       0           0
## GSM1587801                       0           0
## GSM1587802                       0           0
## GSM1587803                       0           0
## GSM1587804                       0           0
```

the first 20 rows of data are gene exprs from heterogeneous mixtures to be deconvolved. The remaining rows are references for each of several PBMC cell types (B, NK, T, etc). Because there are so many cell types we collapse some of the sub-types into a fewer number of general leukocyte types:

```
general_types = factor(sapply(strsplit(colnames(mix)," "),"[",1))
mix = sapply(levels(general_types),function(g)rowSums(mix[,general_types==g,drop=FALSE]))
```

We can extract out which rows are pure reference samples:

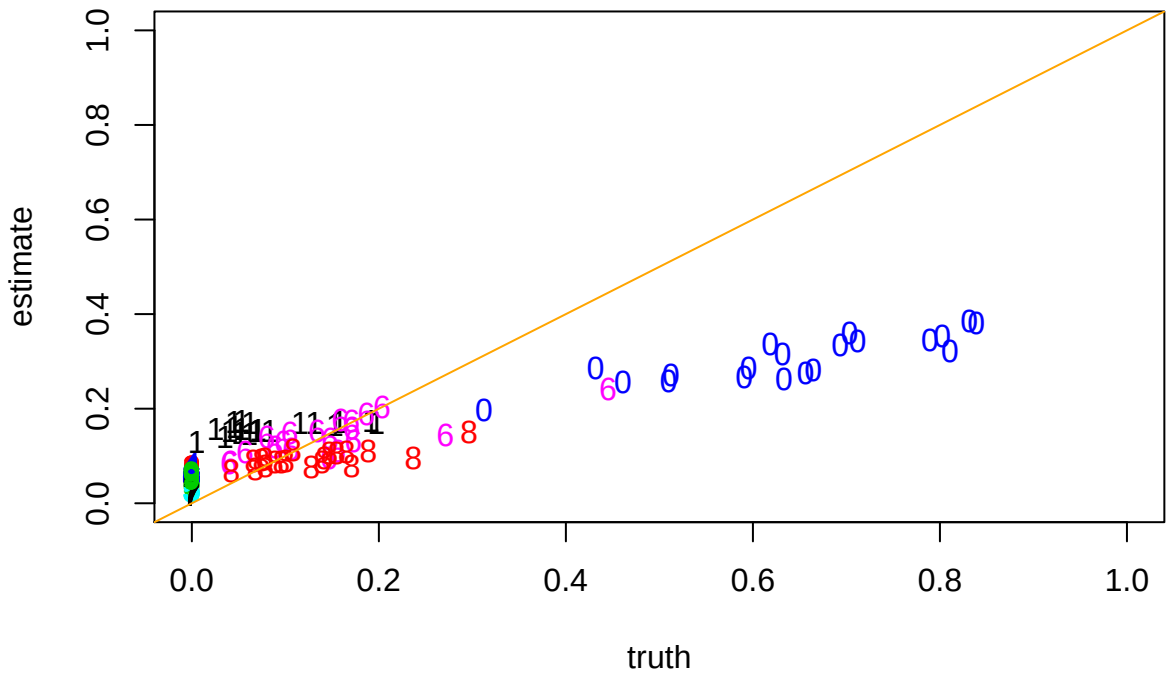
```
pure_samples = lapply(1:ncol(mix),function(i)which(mix[,i]==1))
names(pure_samples) = colnames(mix)
lapply(pure_samples,head,n=2)
```

```
## $B
## Bcell.naive.1..HG.U133A...IRIS_GSE22886.GSM565308.
##                                                 21
## Bcell.naive.2..HG.U133A...IRIS_GSE22886.GSM565309.
##                                                 22
##
## $Dendritic
## DendriticCell.Control.1..HG.U133A...IRIS_GSE22886.GSM565366.
##                                                        108
## DendriticCell.Control.2..HG.U133A...IRIS_GSE22886.GSM565367.
##                                                        109
##
## $Eosinophils
## A_MF_2hrEosinophils_U133A..Chtanova_immune.A_MF_2hrEosinophils_U133A.
##                                                                  124
##       A_MF_ControlEosinophil..Chtanova_immune.A_MF_ControlEosinophil.
##                                                                  125
##
## $Macrophages
## Monocyte.Day7.1..HG.U133A...IRIS_GSE22886.GSM565354.
##                                                  90
## Monocyte.Day7.2..HG.U133A...IRIS_GSE22886.GSM565355.
##                                                  91
##
## $Mast
##       A_LW_mastcellctrl_U133A..Chtanova_immune.A_LW_mastcellctrl_U133A.
##                                                                    120
## A_MF_ControlMASTCELL_U133A..Chtanova_immune.A_MF_ControlMASTCELL_U133A.
##                                                                    121
##
## $Monocytes
## Monocyte.Day0.1..HG.U133A...IRIS_GSE22886.GSM565330.
##                                                  78
## Monocyte.Day0.2..HG.U133A...IRIS_GSE22886.GSM565331.
##                                                  79
##
## $Neutrophils
##   A_LW_neutrophil_U133A..Chtanova_immune.A_LW_neutrophil_U133A.
##                                                            126
## A_MF_neutrophils_U133A..Chtanova_immune.A_MF_neutrophils_U133A.
##                                                            127
##
## $NK
## NKcell.control.1..HG.U133A...IRIS_GSE22886.GSM565293.
##                                                   63
## NKcell.control.2..HG.U133A...IRIS_GSE22886.GSM565294.
##                                                   64
##
## $Plasma
## PlasmaCell.FromPBMC.1..HG.U133A...IRIS_GSE22886.GSM565323.
```

```
##                                                              36
## PlasmaCell.FromPBMC.2..HG.U133A...IRIS_GSE22886.GSM565324.
##                                                              37
##
## $T
## CD8Tcell.N0.1..HG.U133A...IRIS_GSE22886.GSM565269.
##                                                          43
## CD8Tcell.N0.2..HG.U133A...IRIS_GSE22886.GSM565270.
##                                                          44
```

and use those to deconvolve the other samples

```
dt = dtangle(Y=data,pure_samples=pure_samples,n_markers = 100,data_type='microarray-gene')
matplot(mix[-unlist(pure_samples),],dt$estimates[-unlist(pure_samples),],xlab="truth",ylab="estimate",
        ylim=c(0,1),xlim=c(0,1));abline(coef=c(0,1),col='orange')
```



Since the cell types in the mixtures are known in this case we can subset the cell types to only look for to those cell types that we know exist in the data. First we determine what cell types are present and subset the data appropriately,

```
known_types = colnames(mix)[colSums(mix[1:20,])>0]
known_types
```

```
## [1] "B"         "Monocytes" "NK"        "T"
```

```
keep_rows = c(1:20,unlist(pure_samples[known_types]))
data = data[keep_rows,]
mix = mix[keep_rows,known_types]
pure_samples = lapply(1:ncol(mix),function(i)which(mix[,i]==1))
```

```
names(pure_samples) = colnames(mix)
```

and then run dtangle on the subsetted data

```
dt = dtangle(Y=data,pure_samples=pure_samples,n_markers = 100,data_type='microarray-gene')
matplot(mix[-unlist(pure_samples),],dt$estimates[-unlist(pure_samples),],xlab="truth",ylab="estimate",
        ylim=c(0,1),xlim=c(0,1));abline(coef=c(0,1),col='orange')
```