eBass Demo

Yunjiang Ge

12/17/2020

R Markdown

This is an R Markdown document for eBass demo.

Required packages

```
library(R.matlab)
library(locfdr)
```

Import and initialize sample data

```
data <- readMat('/Users/bright1/Downloads/sampledata.mat')
pdata<-data$data2[1:10000]
tdata<-data$data3[1:10000]
```

Step 1: Obtain the empirical Bayes estimation for prior probabilities, non-null and null distributions. The output includes a list of $\hat{z}$ with their corresponding Sensitivity, TDR, Objective Function, T-score, Primary Threshold. Function input: tdata = T-map obtained from the image (either from t-test or regression parameter); pdata = p-values corresponds to T-map; sample size per arm (ssample data set 30 subjects per arm); number of breaks in Bayes two component model estimation. Note that in this demo the sample data is simulated data from Gaussian distribution, using the default value for type=1 MLE can handle most of the situations. In real data analysis, you will need to preview the data and adjust the estimating methods accordingly. Similar for number of breaks and type.

```
ebass_est<-primarythres(tdata,pdata,30,100)
```

Step 2: Prepare the input for estimating the support Ω_α

```
#try to restrict the max search primary threshold to 0.02(empirical value)
and save time
#find out the index
idx<-length(which(ebass_est[,5]<0.02))
sort_p<-sort(pdata)
vox_num<-fdr_cut<-loc<-results<-c()
for (i in 1:idx)
{
vox_num[i]<-length(which(pdata<=ebass_est[i,5]))
loc[i]<-ceiling(vox_num[i]*(1-ebass_est[i,2]))
fdr_cut[i]<-sort_p[loc[i]]
}
```

The results include Primary threshold, the cutoff p-value given by estimated false positive voxels, total number of suprathreshold voxels, total number of false positive suprathreshold voxels.

```
results<-as.data.frame(cbind(ebass_est[1:idx,5], fdr_cut,vox_num,loc))
names(results)<-
c("PrmThres","fdr_cutoff","suprathres_voxnum","false_supra_voxnum")
results
```

```
##        PrmThres   fdr_cutoff suprathres_voxnum false_supra_voxnum
## 1  0.0007215047 2.608452e-07               145                   8
## 2  0.0009933075 4.461816e-07               157                  10
## 3  0.0013595036 1.184183e-06               176                  14
## 4  0.0018494578 4.798388e-06               199                  20
## 5  0.0025003006 1.245613e-05               219                  26
## 6  0.0033584694 2.643482e-05               250                  36
## 7  0.0044813742 4.890435e-05               276                  48
## 8  0.0059391300 8.502069e-05               315                  65
## 9  0.0078162723 2.039453e-04               366                  89
## 10 0.0102133401 4.300681e-04               411                 117
## 11 0.0132481781 8.893468e-04               462                 152
## 12 0.0170567820 1.910899e-03               535                 202
```

Step 3: Next we save the output and transport it to the MATLAB code. Since R markdown doesn't support MATLAB platform yet, we ran the script seperately and save one result to display here. The input values from R is same since the empirical Bayes estimation is not changed, but the cluster size threshold given from permutation test may vary slightly. This is caused by the randomness from permutation test. To test the MATLAB, you simply need to download 'demo_run.m' and execute it.

The following chunk is used to import and re-format MATLAB output. No need to change.

Final Result The output includes the primary threshold, cluster-size threshold, final voxel-level Sentivity and TDR. In addition, the results from 0.001 is provided for comparison.

```
##                    eBass     0.001
## PrmThres     0.001849458  0.001000
## ClusThres   13.000000000 12.000000
## Sensitivity  0.373165618  0.293501
## TDR          1.000000000  1.000000
```