Random pooling algorithm

**Background**

As collecting samples is complicated, time-consuming and the number of the samples is limited, the differentially expressed genes would be greatly related to the sample set. In order to minimize this effect and false positive caused by the special sample set, we developed “random pooling algorithm” combining to “DESeq2” to obtain reliable and stable differentially expressed genes.

**Description and details**

The aim of this algorithm is to calculate the differentially expressed genes based on the re-sampling the samples from the original sample set. The detail of the algorithm is:

1. To estimate the sample sizes of the two groups and select the small size group as the random pooling reference group;
2. To set up *i*, which contains the number of samples that could be selected from group1 and group2, here, *i* is from 3 to the number of samples in random pooling reference group;
3. To randomly select same number *i* of the samples from group1 and group2, respectively;
4. To calculate the differentially expressed genes by using selected samples;
5. To repeat 2)-3) for *j* times.

**Usage and arguments**

The inputs include:

1. Matrix1: the read count matrix of group1, in which the rows are genes and the columns are samples;
2. Matrix2: the read count matrix of group2, in which the rows are genes and the columns are samples;
3. Sn: it represents “j”, which means the repeating times in random pooling process;
4. Group1Inf: a dataframe, containing the group1’s information;
5. Group2Inf: a dataframe, containing the group2’s information;
6. OutDir: the folder to save the results;
7. Parallel\_TF: a logical value. If it is TRUE, the DESeq will run by using BiocParallel; if it is FALSE, the DESeq will run by single core;
8. Ncores: the number of cores used when running DESeq;