

# iECAT Package

Seunggeun (Shawn) Lee

May 10, 2017

## 1 Overview

iECAT package has functions to test for associations between sets of variants and phenotypes with integrating external control samples. The main function, iECAT, is similar to the SKAT function in the SKAT package but needs addition information on allele counts of the external control samples. To use this package, SKAT and MetaSKAT packages should be installed.

## 2 Association test

### 2.1 Example

An example dataset (`Example`) has a list of ten genotype matrices (`Z.list`), a vector of binary phenotype (`Y`), a list of tables for external control samples.

```
> library(SKAT)
> library(iECAT)
> data(Example)
> attach(Example)
```

As the same as the SKAT function, the `SKAT_Null_Model` function (in SKAT package) should be used in prior to carry out association tests. `SKAT_Null_Model` estimates parameters under the null model and obtain residuals. The following code performs iECAT-O, SKAT-O version of iECAT, with incorporating allele count information in `tbl.external.all.list`.

```
> # iECAT-O
> # test the first gene
>
> obj<-SKAT_Null_Model(Y ~ 1, out_type="D")
> Z = Z.list[[1]]
> tbl.external.all = tbl.external.all.list[[1]]
> out<-iECAT(Z, obj, tbl.external.all, method="optimal")
> #iECAT-O pvalue
> out$p.value
```

```

[1] 2.246796e-10
> # p-value without batch-effect adjustment
> out$p.value.noadj
[1] 3.575066e-12
> # p-value computed without using external control samples. SKAT-0 is used to compute this
> out$p.value.internal
[1] 3.064378e-06
>

```

To run SKAT type dispersion test ( $\rho = 0$ ) or Burden type collapsing test ( $\rho=1$ ), you need to specify `r.corr=0` or `r.corr=1`.

```

> # rho=0
> iECAT(Z, obj, tbl.external.all, r.corr=0)$p.value
[1] 4.265712e-08
> # rho=1
> iECAT(Z, obj, tbl.external.all, r.corr=1)$p.value
[1] 5.377425e-08
>

```

As the same as in the SKAT function, the `weights.beta` parameter can be changed. Users can also specify their custom weights (`weight`). Users can also specify the upper limit of MAFs. For example, if users want to test for associations with variants with  $MAF < 0.01$ , `MAF.limit` should be 0.01

```

> # test for rare variants only
> iECAT(Z, obj, tbl.external.all, MAF.limit=0.01)$p.value
[1] 1.240584e-24

```

## 2.2 Plink Binary format files

iECAT package can use plink binary format files for genome-wide data analysis. To use plink files, users should provide plink binary files (`bed`, `bim`, `fam`); `setid` file which has set ids and variant list of each set; and external control allele count file (EC file), which has allele counts of each variants in external control samples. The EC file should have six columns: Chromosome, base-pair position, `SNP_ID`, `Allele1`, `Allele2`, `Allele1 count`, `Allele 2 count`. It should not have header. Below is the example:

```

1 10001 snp_10001 A T 39 19961
1 10002 snp_10002 G T 27 19973
: : : : : :

```

Example files can be found at <https://sites.google.com/a/umich.edu/leeshawn/software>.

```

> # To run this code, first download and unzip example files
>
> #####
> #          Generate SSD file
>
> # Create the MW File
>
>
> File.Bed<-"./iECAT.example.bed"
> File.Bim<-"./iECAT.example.bim"
> File.Fam<-"./iECAT.example.fam"
> File.EC<-"./iECAT.example.EC"
> File.SetID<-"./iECAT.example.SetID"
> File.SSD<-"./iECAT.SSD"
> File.Info<-"./iECAT.SSD.INFO"
> File.EC.Info<-"./iECAT.SSD.ECINFO"
> FAM<-Read_Plink_FAM(File.Fam, Is.binary=TRUE)
> # 5 SNP sets in the example in the vignette
> Generate_SSD_SetID_wEC(File.Bed, File.Bim, File.Fam, File.SetID, File.EC, File.SSD, File.I

227 SNPs observed in both SetID (and BIM) and EC files
227 After checking alleles, 227 SNPs left

Check duplicated SNPs in each SNP set
No duplicate
10000 Samples, 5 Sets, 227 Total SNPs
[1] "SSD and Info files are created!"

> EC.INFO= Open_SSD_wEC(File.SSD, File.Info, File.EC.Info)

10000 Samples, 5 Sets, 227 Total SNPs
Open the SSD file

> obj<-SKAT_Null_Model(Phenotype ~ 1, out_type="D", data=FAM)
> re<-iECAT.SSD.All(EC.INFO, obj=obj, method="optimal")
> re

$results
      SetID      P.value P.value.Noadj P.value.Internal N.Marker.All
1 SET_0001 2.246796e-10 3.575066e-12 3.064378e-06 53
2 SET_0002 8.332541e-36 2.462946e-40 2.321727e-20 40

```

```
3 SET_0003 7.203494e-03 2.590867e-03 2.427129e-03 38
4 SET_0004 7.582777e-12 1.310897e-11 2.108541e-07 43
5 SET_0005 6.497645e-19 1.988825e-23 1.981221e-13 53
```

```
N.Marker.Test
```

```
1 53
2 40
3 38
4 43
5 53
```

```
$P.value.Resampling
```

```
NULL
```

```
attr("class")
```

```
[1] "SKAT_SSD_ALL"
```

```
>
```