IGREX for quantifying the impact of genetically regulated expression on phenotypes

CAI Mingxuan

2019/9/12

Introduction

This vignette provides an introduction to the `iGREX` package. R package `iGREX` implements IGREX, a statistical model for quantifying the impact of genetically regulated expression on complex traits and diseases. The package can be installed with the command:

```
library(devtools)
install_github("mxcai/iGREX")
```

The package can be loaded with the command:

```
library("iGREX")
```

Fit IGREX using simulated data

We first demonstrate the workflow of IGREX using simulated data. The genotype matrix can be simulated using `genRawGeno` function:

```
library(mvtnorm)
set.seed(10)
n1 <- 1000
n2 <- 4000
p <- 100   #number of SNPs in each gene
G <- 200   #number of genes
maf <- runif(p*G, 0.05, 0.5)
rho <- 0.8
X <- genRawGeno(maf,p,G,rho,n1+n2)
X1 <- X[1:n1,]
X2 <- X[(n1+1):(n1+n2),]
```

Then, cis effect sizes are generated from standard Gaussian distribution with sparse structure. We set only 20% of cis effects as non-zero, and control the cellular level heritability at $\text{PVE}_y = 0.3$.

```
sb2_true <- 0.3
sy2_true <- 0.7
Y0 <- matrix(0,n1+n2,G)
for(g in 1:G){
  nonzero <- rbinom(p,1,0.2)
  if(sum(nonzero)==0) nonzero[1] <- 1
  beta <- rnorm(p,0,sqrt(sb2_true/sum(nonzero)))
  beta[nonzero==0] <- 0
  Y0[,g] <- X[,((g-1)*p+1):(g*p)] %*% beta
}
Y <- Y0[1:n1,] + matrix(rnorm(n1*G,0,sqrt(sy2_true)),n1,G)
```

Finally, the phenotype data is generated following the generative model of IGREX with variance components specified as $\text{PVE}_{\text{GREX}} = 0.2$, $\text{PVE}_{\text{Alternative}} = 0.3$. GREX effects are also sparse with only 20% non-zero:

```
sg2_true <- 0.2
sa2_true <- 0.3
t <- mean(diag(Y0[(n1+1):(n1+n2),]%*%t(Y0[(n1+1):(n1+n2),])))
nonzero <- rbinom(G,1,0.2)
if(sum(nonzero)==0) nonzero[1] <- 1
alpha <- as.matrix(rnorm(G,0,sqrt(sg2_true/t*G/sum(nonzero))))
alpha[nonzero==0] <- 0
z0 <- Y0[(n1+1):(n1+n2),] %*% alpha
gamma <- as.matrix(rnorm(p*G,0,sqrt(sa2_true/(p*G))))
z1 <- X2%*%gamma
z <- z0 + z1 + rnorm(n2,0,sqrt(var(z0+z1)))
```

At stage one, IGREX solves G linear mixed models using PX-EM algorithm, obtains the posterior distribution of β_g's and computes K_{GREX} and $K_{Alternative}$.

```
X <- scale(X)
X1 <- X[1:n1,]
X2 <- X[(n1+1):(n1+n2),]
K <- K0 <- 0
for(g in 1: G){
  y_g <- Y[,g]
  X1tmp <- X1[,((g-1)*p+1):(g*p)]
  X2tmp <- X2[,((g-1)*p+1):(g*p)]
  W1 <- matrix(1,n1,1)
  fit_g <- iGREX_Kg(y_g,X1tmp,X2tmp,W1,1e-5,500)
  K <- K + fit_g$K_g
  K0 <- K0 + fit_g$K_g0
}
Ka <- X2 %*% t(X2) / ncol(X2)
K <- K/mean(diag(K))
K0 <- K0/mean(diag(K0))
```

At stage two, IGREX estimates PVE_{GREX} using REML or MoM. Here we perform both REML-based and MoM-based IGREX analysis. The REML0 and MoM0 corresspond to the IGREX analyses without accounting for uncertainty in predicted gene expression. Please note that the MM algorithm for REML takes about 13 minutes on a macOS platform with Intel Core i5 and 8GB RAM. The timing can vary in different platforms. Besides, we set `verbose=F` to turn off the output of algorithm progress in REML, which makes the vignette uncluttered. One can easily track the algorithm progress by setting `verbose=T` in the `REML_3var` function. The MoM-based method only takse seconds.

```
# REML
REML <- REML_3var(K,Ka,z,verbose=F)

# REML without acocunting for uncertainty
REML0 <- REML_3var(K0,Ka,z,verbose=F)

# exact estimate by MoM
MoM <- MoM_3var(K,Ka,z)

# exact estimate by MoM without accounting for uncertainty
MoM0 <- MoM_3var(K0,Ka,z)
```

The $\widehat{\text{PVE}}_{\text{GREX}}$ and $\widehat{\text{PVE}}_{\text{Alternative}}$ as well as their standard errors are returned:

```
REML$PVE
```

```
##                    PVEg       PVEa         h2       prop
## heritability 0.19497625 0.32108764 0.51606388 0.37781416
## se           0.02396532 0.03235716 0.03187339 0.04393594
```

```
REML0$PVE
```

```
##                    PVEg       PVEa         h2       prop
## heritability 0.13598493 0.35686646 0.49285139 0.27591466
## se           0.01679333 0.03107969 0.03120717 0.03371781
```

```
MoM$PVE
```

```
##                    PVEg       PVEa         h2       prop
## heritability 0.22730331 0.32224632 0.54954963 0.41361743
## se           0.04097663 0.05678818 0.05166199 0.07438731
```

```
MoM0$PVE
```

```
##                    PVEg       PVEa         h2       prop
## heritability 0.15762794 0.35850286 0.51613081 0.30540309
## se           0.02842093 0.05410076 0.05059697 0.05850694
```

Fit iGREX using individual level genotype and eQTL data

Please note that the analysis in this section relies on the NFBC dataset and the GTEx dataset. One may not be able to follow the procedure if these datasets are not available. We provide an example of fitting iGREX using GWAS and eQTL data in plink binary format. To use the function, we first specify the input file names:

```
file_qtlgeno <-  "GTEx_qc_hm3"
file_GWASgeno <- "NFBC_filter_mph10"
file_expression <- "Liver_gene_expression.txt"
file_qtlcov <- "Liver_cov.txt"
file_GWAScov <- "NFBC_hm3_filter.eigenvec"
load("K_NFBC_hm3.RData")
```

Here, `file_qtlgeno` is the prefix for eQTL genotype data in plink binary format, `file_GWASgeno` is the GWAS data in plink binary format, `file_expression` is the gene expression file with extended name, `file_qtlcov` and `file_GWAScov` are covariates file for eQTL and GWAS data, respectively. For gene expresion file, it must have the following format (rows for genes and columns for individuais and note that it must be tab delimited):

lower	up	genetype1	genetype2	TargetID	Chr	HG00105	HG00115
59783540	59843484	lincRNA	PART1	ENSG00000152931.6	5	0.5126086	0.7089508
48128225	48148330	protein_coding	UPP1	ENSG00000183696.9	7	1.4118007	-0.0135644
57846106	57853063	protein_coding	INHBE	ENSG00000139269.2	12	0.5755268	-1.0162217
116054583	116164515	protein_coding	AFAP1L2	ENSG00000169129.8	10	1.1117776	0.0407033
22157909	22396763	protein_coding	RAPGEF5	ENSG00000136237.12	7	0.2831573	-0.1772559
11700964	11743303	lincRNA	RP11-434C1.1	ENSG00000247157.2	12	0.2550282	-0.2831573

The covariate files must have the following format (rows for individuals and the first two columns are FID and IID):

V1	V2	V3	V4	V5	V6	V7	V8
GTEX-11DXY	GTEX-11DXY	0.0109	-0.0027	-0.0789	-0.0552724	0.0469389	0.0488902
GTEX-11DXZ	GTEX-11DXZ	-0.1053	-0.0126	0.0027	-0.0097383	0.0542790	-0.0918273
GTEX-11EQ9	GTEX-11EQ9	0.0148	-0.0078	-0.0057	0.0362291	0.0474735	-0.0502703

V1	V2	V3	V4	V5	V6	V7	V8
GTEX-11GSP	GTEX-11GSP	0.0168	-0.0052	-0.0019	-0.0505170	-0.0960899	-0.0554980
GTEX-11NUK	GTEX-11NUK	0.0154	-0.0025	0.0049	-0.0733902	0.0009193	0.0227932
GTEX-11NV4	GTEX-11NV4	-0.1152	-0.0170	-0.0223	-0.0619178	0.0035294	0.0361403

The alternative genetic correlation matrix (`Ka`) is just the kinship matrix and should be manually computed (which can be done by plink or GEMMA) and supplied to the function. Finally, the analysis can be done by `iGREX` function:

```
fit <- iGREX(file_qtlgeno,file_GWASgeno,file_expression,file_qtlcov,file_GWAScov,Ka=K,
             whCol=6,bw=500000,subsample = 0,method="MoM")
```

The estimated PVEs are stored in the outcome list and can be accessed by (`fit$output$PVE`)

	PVEg	PVEa	h2	prop
heritability	0.1917895	0.4069149	0.5987045	0.3203409
se	0.0408129	0.1001861	0.0937858	0.0823769

Fit iGREX-s using GWAS summary statistics and eQTL data

Next, we provide an example of fitting iGREX-s using GWAS summary statistics and eQTL data with an LD reference. To use the function, we first specify the input file names:

```
file_qtlgeno =  file_GWASgeno = "IGREX_data/all_chr_1000G"
file_expression = "IGREX_data/Geuvadis_gene_expression_qn.txt"
file_qtlcov = file_GWAScov = "IGREX_data/TGP_ref_pc20.eigenvec"
file_z = "IGREX_data/NFBC_HDL_cov20_summary.txt"
load("IGREX_data/K_TGP_height.RData")
```

In this function, `file_qtlgeno`, `file_GWASgeno`, `file_expression`, `file_qtlcov` and `file_GWAScov` are of the same format as the ones in `iGREX`, while an additional GWAS summary ststistics file `file_z` is needed. All the data here are open access: the expression data is from the Geuavadis project, whose samples are from 1000 Genomes Europe ancestry; we use the genotypes of these 1000 Genomes samples as the LD reference panel. All files used here can be downloaded from this Dropbox link. The summary statistics should be of the following format:

SNP	N	Z	INC_ALLELE	DEC_ALLELE
cnv3p502	5123	-1.3533719	A	G
rs3115860	5123	0.5287950	C	A
rs3131967	5123	0.7043162	A	G
rs12124819	5123	-0.3059190	G	A
rs17160939	5123	1.2516159	G	A
rs4475691	5123	1.1862824	A	G

The IGREX-s can then be fitted using `iGREXs` function:

```
fit_s <- iGREXs(file_qtlgeno,file_GWASgeno,file_expression,file_z,file_qtlcov,file_GWAScov,
                Ka=K,bw = 500000)
```

The estimated PVEs are stored in the outcome list and can be accessed by `fit_s$output$PVE`

	PVEg	PVEa	prop
heritability	0.0727114	0.4729560	0.1332522
se	0.0435076	0.1177819	0.0764512