GWAS 2

Jinliang Yang

04-28-2022

Path Normalization

LD Decay

Linkage disequilibrium (LD) refers to the nonrandom associations of alleles at different loci. - The decay of LD is affected by recombination rate and the number of generations of recombination. Therefore, investigating LD decay may reveal the population recombination history. - For GWAS, it helps to estimate the number of markers needed.

–

using PLINK to compute LD

- With `--r2`, when a table format report is requested, pairs with r2 values less than 0.2 are normally filtered out of the report.
- Use `--ld-window-r2` to adjust this threshold.

LD Decay

- Using HCC to conduct the analysis

```
### log onto HCC
ssh YOUR_USER_ID@crane.unl.edu
# Enter your passcode

### request computing node
srun --qos=short --nodes=1 --licenses=common --ntasks=4 --mem 32G --time 6:00:00 --pty bash
# cd to your project repo
# git pull
```

- Load the `PLINK` module

```
module load plink
cd largedata/RiceDiversity_44K_Genotypes_PLINK/

# By default, when a limited window report is requested,
# every pair of variants with at least (10-1) variants between them,
# or more than 1000 kilobases apart, is ignored.
# You can change the first threshold with `--ld-window`, and the second threshold with `--ld-window-kb`
plink -bfile binary_sativas413 --r2 --ld-window 100 --ld-window-kb 100 --ld-window-r2 0 --out binary_sa
```

Summarize LD decay rate

```
library("data.table")
# cd back to your project home dir
df <- fread("largedata/RiceDiversity_44K_Genotypes_PLINK/binary_sativas413.ld", data.table=FALSE)

BINSIZE = 100
df$dist <- df$BP_B - df$BP_A
df$bin <- round(df$dist/BINSIZE, 0)

library(plyr)

df2 <- ddply(df, .(bin), summarise,
      meanr2 = mean(R2))

write.table(df2, "cache/ld_in_100bp_bin.csv", sep=",", row.names=FALSE, quote=FALSE)
```

Plot LD decay results

- Plot the figure and sync the figure through github

```
ld <- read.csv("cache/ld_in_100bp_bin.csv")

pdf("graphs/ld_decay.pdf", width=10, height=10)
plot(ld$bin*100, ld$meanr2, xlab="Physical distance (bp)", ylab="R2", main="LD decay rate in rice")
abline(h=0.3, col="red")
dev.off()
```

_

- Sync the results through github and plot the figure on local computer

```
ld <- read.csv("cache/ld_in_100bp_bin.csv")

plot(ld$bin*100, ld$meanr2, xlab="Physical distance (bp)", ylab="R2", main="LD decay rate in rice")
abline(h=0.3, col="red")
```

Population structure using PCA

PCA (principal component analysis) is a method often used to compress the high dimensional data without losing as much information.

Basically, it creates linear combinations of the columns of matrix information, **X**, and generates, at most, p linear combinations, called principal components.

$$
\begin{aligned}
PC_1 &= \mathbf{w_1 X} \\
PC_2 &= \mathbf{w_2 X} \\
PC_p &= \mathbf{w_p X}
\end{aligned}
$$

Here, $\mathbf{w_p}$ is the **eigenvector** of PC_p.

The first PC, or PC1, captures the largest variance, the 2nd PC, or PC2, captures the 2nd largest variance, and so on.

PCA using PLINK

- By default, `--pca` extracts the top 20 principal components; you can change the number by passing a numeric parameter.
- Eigenvectors are written to `plink.eigenvec`, and top eigenvalues are written to `plink.eigenval`.
- The 'header' modifier adds a header line to the `.eigenvec` file(s).

```
cd largedata/RiceDiversity_44K_Genotypes_PLINK
plink -bfile binary_sativas413 --pca 'header' --out sativas413
cd ../../
cp largedata/RiceDiversity_44K_Genotypes_PLINK/sativas413.eigenvec cache/
```

PCA using PLINK

Plot the PCA results

```
pca <- read.table("cache/sativas413.eigenvec", header=TRUE)
plot(pca$PC1, pca$PC2, xlab="PC1", ylab="PC2")
plot(pca$PC3, pca$PC4, xlab="PC3", ylab="PC4")
```

```
# install.packages("scatterplot3d")
library("scatterplot3d")

fsize=16
pdf("graphs/pca_3d.pdf", width=10, height=10)
scatterplot3d(pca[,3:5], pch = 16, cex.symbol=1.2, color="#00BFC4", main="Maize Diversity Panel", angle=
dev.off()
```

GWAS using the gemma software package

Fit the QK model

$$\mathbf{y} = \mathbf{Qv} + \mathbf{w_i} m_i + \mathbf{Zu} + \mathbf{e}$$

```
module load gemma
# To calculate centered relatedness matrix (will take ~ 1 min):
gemma -bfile binary_sativas413 -gk 1 -o binary_sativas413
```

–

-9 Phenotype doesn't go well with gemma. We have to change a little bit.

GWAS using the gemma software package

Fit the QK model

```
library("data.table")

ped <- fread("sativas413.ped", header=FALSE)
ped$V6 <- 1
fwrite(ped, "sativas413.ped", sep="\t", row.names=FALSE, col.names = FALSE, quote=FALSE)

fam <- fread("sativas413.fam", header=FALSE)
fam$V6 <- 1
fwrite(fam, "sativas413.fam", sep="\t", row.names=FALSE, col.names = FALSE, quote=FALSE)
```

–

```
module plink
plink --file sativas413 --make-bed --out binary_sativas413
# To calculate centered relatedness matrix (will take ~ 1 min):
gemma -bfile binary_sativas413 -gk 1 -o binary_sativas413
```

```
library("data.table")
k <- fread("largedata/RiceDiversity_44K_Genotypes_PLINK/output/binary_sativas413.cXX.txt", header=FALSE
dim(k)
```

GWAS using the gemma software package

Q matrix

If one has covariates other than the intercept and wants to adjust for those covariates simultaneously, one should provide GEMMA with a covariates file containing an intercept term explicitly. > from Gemma manual

```
1 1 -1.5
1 2 0.3
1 2 0.6
1 1 -0.8
1 1 2.0
```

_

```
# cd to largedata/RiceDiversity_44K_Genotypes_PLINK
pca <- read.table("sativas413.eigenvec", header=TRUE)
pca[,2] <- 1
write.table(pca[,2:5], "pc3.txt", sep="\t", row.names=FALSE,
            quote=FALSE, col.names = FALSE)
```

Phenotypic data

```
pheno <- read.delim("http://ricediversity.org/data/sets/44kgwas/RiceDiversity_44K_Phenotypes_34traits_P

library(ggplot2)
ggplot(pheno, aes(x=Plant.height)) +
  geom_histogram(aes(y=..density..), bins=50, fill="#999999")+
  geom_density(alpha=.2, fill="#FF6666") +
  labs(title="Phenotype histogram plot",x="Plant Height", y = "Density")+
  theme_classic()
```

_

- On HCC, write the `pheno.txt` to the genotypic data folder

```
pheno <- read.delim("http://ricediversity.org/data/sets/44kgwas/RiceDiversity_44K_Phenotypes_34traits_P
write.table(pheno[, -1:-2], "largedata/RiceDiversity_44K_Genotypes_PLINK/pheno.txt",
            sep="\t", row.names=FALSE, quote=FALSE, col.names = FALSE)
```

GWAS using the gemma software package

$$\mathbf{y} = \mathbf{Q}\mathbf{v} + \mathbf{w_i} m_i + \mathbf{Z}\mathbf{u} + \mathbf{e}$$

```
gemma -bfile binary_sativas413 -c pc3.txt -k output/binary_sativas413.cXX.txt -p pheno.txt -lmm 4 -n 12

cp output/Plant.height.assoc.txt ../../cache
```

- `lmm`: specify frequentist analysis choice (default 1; valid value 1-4; 1: Wald test; 2: likelihood ratio test; 3: score test; 4: all 1-3.)
- `n`: specify phenotype column in the phenotype file (default 1); or to specify which phenotypes are used in the mvLMM analysis
- `o`: specify output file prefix
- `miss`: specify missingness threshold (default 0.05)

- `r2`: specify r-squared threshold (default 0.9999)
- `hwe`: specify HWE test p value threshold (default 0; no test)
- `maf`: specify minor allele frequency threshold (default 0.01)

The Manhattan plot

```
library(qqman)
library("data.table")
res <- fread("cache/Plant.height.assoc.txt")

manhattan(x = res, chr = "chr", bp = "ps", p = "p_wald", snp = "rs", col = c("blue4", "orange3"), logp =
```

---

rrBLUP package for GWAS

PLINK format

```
# install.packages("BGLR")
library("BGLR")
ped <- read_ped("data/RiceDiversity_44K_Genotypes_PLINK/sativas413.ped")
```

Recoding the data to `0,1,2`

```
p=ped$p
n=ped$n
out=ped$x
#Recode snp to 0,1,2 format using allele 1
# 0 --> 0
# 1 --> 1
# 2 --> NA
# 3 --> 2
out[out==2]=NA
out[out==3]=2
Z <- matrix(out, nrow=p, ncol=n, byrow=TRUE)
Z <- t(Z)
dim(Z) # # 413 x 36901
```

---

rrBLUP package for GWAS

Read fam file

```
# accession ID
fam <- read.table("data/RiceDiversity_44K_Genotypes_PLINK/sativas413.fam", header = FALSE, stringsAsFac
head(fam)
rownames(Z) <- paste0("NSFTV_", fam$V2) # 413 x 36901
```

SNP imputation

```
for (j in 1:ncol(Z)){
  Z[,j] <- ifelse(is.na(Z[,j]), mean(Z[,j], na.rm=TRUE), Z[,j])
}
```

rrBLUP package for GWAS

```
# install.packages("rrBLUP")
library(rrBLUP)
map <- read.table("data/RiceDiversity_44K_Genotypes_PLINK/sativas413.map", header = FALSE, stringsAsFact
mygeno <- data.frame(marker=map[,2], chrom=map[,1], pos=map[,4], t(Z-1), check.names = FALSE) # Z = \in
pheno <- read.delim("http://ricediversity.org/data/sets/44kgwas/RiceDiversity_44K_Phenotypes_34traits_PL

pheno$NSFTVID <- paste0("NSFTV_", pheno$NSFTVID)
mypheno <- data.frame(NSFTV_ID=pheno$NSFTVID, y=pheno$Plant.height)

res2 <- GWAS(mypheno, mygeno, fixed=NULL, n.PC=3, min.MAF=0.05, P3D=TRUE, plot=FALSE)

# Returns a data frame where the first three columns are the marker name, chromosome, and position, and
library(qqman)
pdf("graphs/mht_res2.pdf", width=10, height=5)
manhattan(x = res2, chr = "chrom", bp = "pos", p = "y", snp = "marker", col = c("blue4", "orange3"), log
dev.off()
```