BMP4 in the European sample: A pipeline for targeted sequence analysis

Samuel G. Younkin

December 12, 2012

```
> library("vcf2R")
> library("trioClasses")
> library("trio")
> data("8q24-european-all.geno")
> data(ped, package = "trioClasses")
```

After loading the necessary packages and data we first make sure that the pedigree data frame contains fields F, M and O for father, mother and offspring ids. Note that these ids should match those in the vcf file.

```
> pedigreeInfo <- within(ped.df, {
     F <- as.character(fid)
     M <- as.character(mid)
     O <- as.character(id)
 })
> tg.ped <- Pedigree(pedigreeInfo = pedigreeInfo)
> tg.ped

This pedigree object contains 1812 trios.
For access to the data frame use the trios() accessor function.
```

Now we do our best to retrieve ids from the vcf/geno data and manipulate them to match the pedigree file.

```
> id.vec <- colnames(geno.mat)
> head(id.vec)

[1] "H_ME-DS11103_02-DS11103_02" "H_ME-DS11103_03-DS11103_03"
[3] "H_ME-DS11107_02-DS11107_02" "H_ME-DS11107_03-DS11107_03"
[5] "H_ME-DS11107_01-DS11107_01" "H_ME-DS11108_02-DS11108_02"

> foo <- strsplit(x = id.vec, split = "-")
> id.vec <- as.character(data.frame((do.call("rbind", foo)))[,
     3])
> length(id.vec)

[1] 218

> head(id.vec)

[1] "DS11103_02" "DS11103_03" "DS11107_02" "DS11107_03" "DS11107_01"
[6] "DS11108_02"
```

There are 218 subjects in the genotpye matrix. There is a problem here, as we expect may more European subjects to exist in the vcf file. But we move on, and now with ids formatted properly we create the genotype object needed for the gTrio class.

```
> genomat <- t(ifelse(geno.mat == "0/0", 0L, ifelse(geno.mat ==
     "0/1", 1L, ifelse(geno.mat == "1/1", 2L, NA))))
> colnames(genomat) <- rownames(geno.mat)
> rownames(genomat) <- id.vec
> geno.trio <- genoMat(tg.ped, genomat)
```

As well as the accompanying Pedigree object, such that all trio memebrs have data in the genotype matrix. These values may all be NA, as NA is not precluded from the genotype matrix.

```
> (tg.ped.complete <- completeTrios(tg.ped, colnames(geno.trio)))

This pedigree object contains 65 trios.
For access to the data frame use the trios() accessor function.
```

Now we create the map data frame that contains chromsome and position information for each SNP.

```
> foo <- strsplit(x = rownames(geno.trio), split = ":")
> map.df <- data.frame((do.call("rbind", foo)))
> names(map.df) <- c("chr", "pos")
```

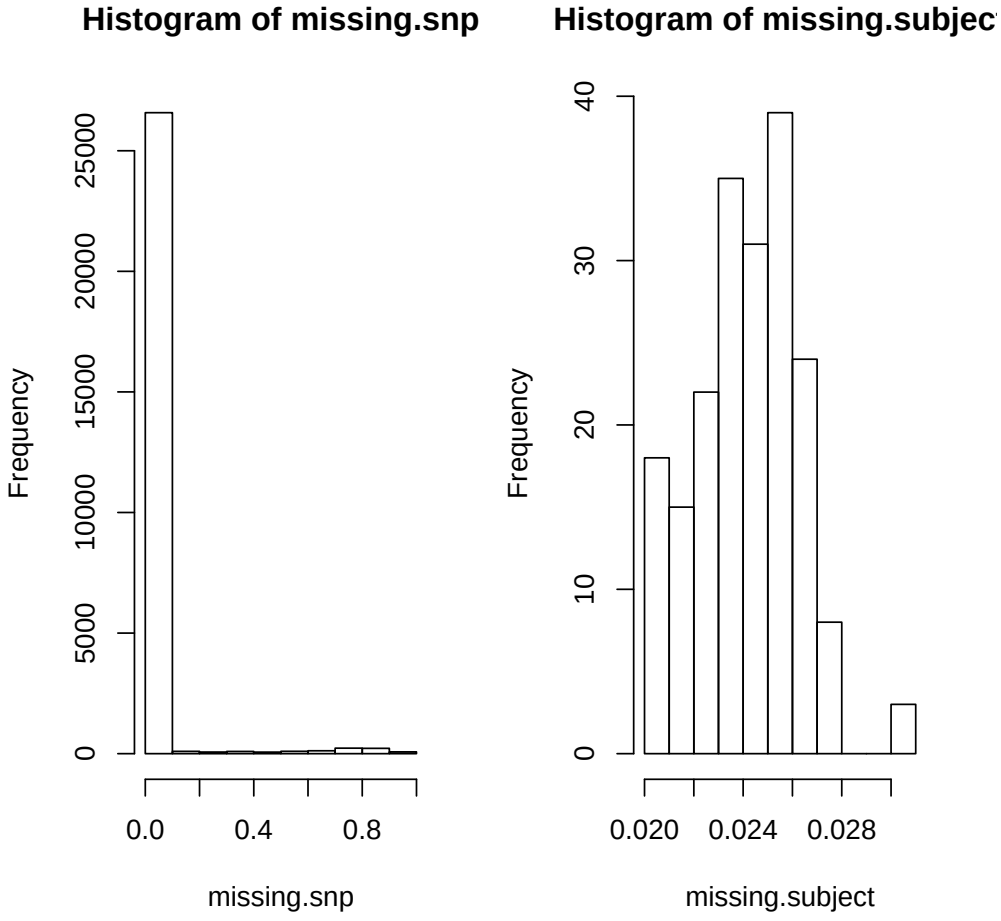
Now we create the gTrio object.

```
> (gTrio.obj <- gTrio(tg.ped.complete, geno = geno.trio,
     map = map.df))

This object has 65 trios and 27639 markers, and a map data frame (with 27639 rows) that starts like this
  chr       pos
1   8 129295457
2   8 129295477
3   8 129295502
4   8 129295529
5   8 129295585
6   8 129295625
```

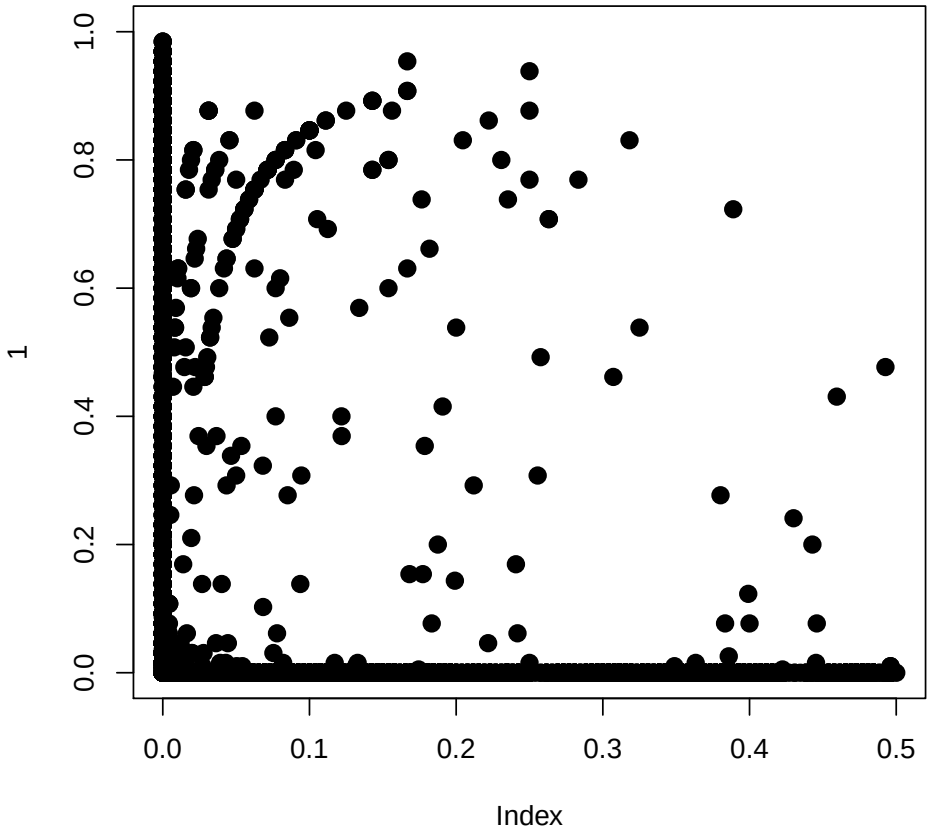
65 trios

```
> missing.snp <- rowSums(is.na(geno(gTrio.obj)))/dim(geno(gTrio.obj))[2]/dim(geno(gTrio.obj))[3]
> missing.subject <- colSums(is.na(geno(gTrio.obj)))/dim(geno(gTrio.obj))[1]
> length(missing.subject)

[1] 195

> length(missing.snp)

[1] 27639


> geno <- geno(gTrio.obj)
> maf <- getMAF(gTrio.obj)
> summary(maf)
```

```
> layout(matrix(1:2, ncol = 2, nrow = 1))
> hist(missing.snp, breaks = 10)
> hist(missing.subject, breaks = 10)
```



```
> plot(1, type = "n", ylim = c(0, 1), xlim = c(0, 0.5))
> points(x = maf, y = missing.snp, pch = 20, cex = 2)
```

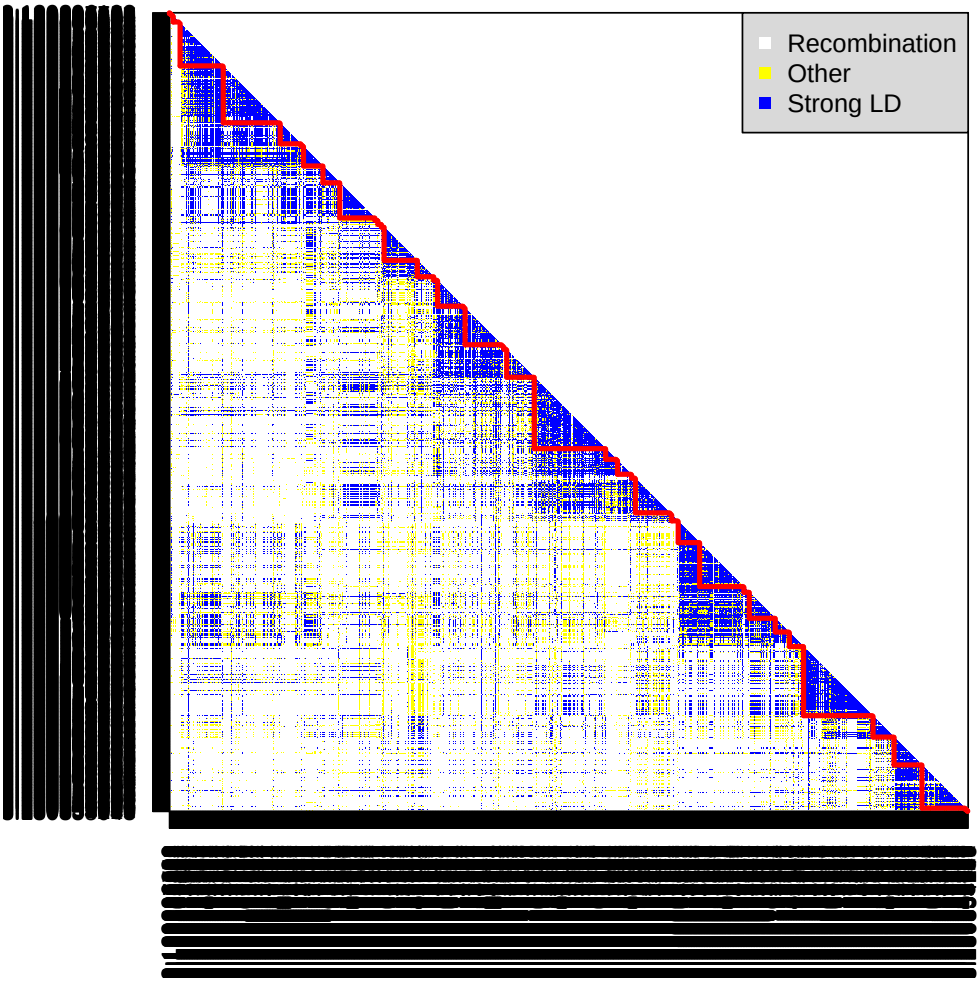


```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
0.00000 0.00000 0.00000 0.01929 0.00000 0.50000       1


> filter <- c(which(missing.snp > 0.1), which(maf < 0.05))
> geno.trio.2 <- geno.trio[-filter, ]
> map.df.2 <- map.df[-filter, ]
> (gTrio.obj <- gTrio(tg.ped.complete, geno = geno.trio.2,
     map = map.df.2))

This object has 65 trios and 2092 markers, and a map data frame (with 2092 rows) that starts like this:
   chr       pos
12   8 129296000
20   8 129296198
35   8 129296750
49   8 129297464
51   8 129297518
62   8 129297807

> geno <- getGeno(gTrio.obj, type = "holger")

> geno <- geno(gTrio.obj)
> maf2 <- getMAF(gTrio.obj)
> summary(maf2)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0500  0.1308  0.1808  0.2326  0.3491  0.5000
```

Now, it's easy to perform any method in Holger's trio package, such as alllelicTDT.

```
> geno.mat <- getGeno(gTrio.obj, type = "holger")
> (aTDT <- allelicTDT(mat.snp = geno.mat, size = 10000))

      Allelic TDT

Top 5 SNPs:
            Statistic   p-value
8:129836588     17.00 3.738e-05
8:129870478     16.25 5.540e-05
8:129842198     15.06 0.0001042
8:129846919     15.06 0.0001042
8:129854330     15.06 0.0001042

> (gTDT <- tdt(snp = geno.mat))

        Genotypic TDT Based on 3 Pseudo Controls

Model Type: Additive

   Coef   OR Lower Upper       SE Statistic   p-Value
 0.1137 1.12 1.106 1.135 0.006723     286.2 < 2.2e-16
```

```
> ld <- getLD(x = geno.mat, which = "both", parentsOnly = TRUE,
     addVarN = TRUE)
> ldblocks <- findLDblocks(ld)
> plot(ldblocks)
```



```
> hist(maf, breaks = 10)
```