

The R package *trioClasses* for definition of the class *SNPTrioExperiment*, an extension of *SummarizedExperiment*, for use in trio based analyses of genetic data.

Samuel G. Younkin

March 13, 2013

1 Packages & Data

```
> library("trioClasses")
> data("sample")
> data("8q24-european-all.sm")
```

2 SummarizedExperiment

```
> se <- SummarizedExperiment(assays = SimpleList(geno = t(sm)),
  colData = col.DF, rowData = gr)
```

3 Pedigree

```
> ped <- PedClass(ped.DF)
```

4 SNPTrioExperiment

```
> (ste <- SNPTrioExperiment(se, pedigree = ped))
```

```
class: SNPTrioExperiment
dim: 8951 960
exptData(0):
assays(1): geno
rownames(8951): chr8:129296000 chr8:129296113 ...
  chr8:130354703 chr8:130354790
rowData metadata column names(0):
colnames(960): H_ME-DS10776_2-DS10776_2
  H_ME-DS10776_3-DS10776_3 ... H_ME-DS11313_3-DS11313_3
  H_ME-DS11313_1-DS11313_1
colData names(1): id
pedigree(4139): famid id fid mid sex dx
complete trios(320):
```

5 Methods

5.1 ScanTrio

```
> (ste.rare <- ste[!(MAF(ste) >= 0.01 | is.na(MAF(ste))))])
```

```
class: SNP Trio Experiment
dim: 6397 960
exptData(0):
assays(1): geno
rownames(6397): chr8:129296113 chr8:129296185 ...
               chr8:130354703 chr8:130354790
rowData metadata column names(0):
colnames(960): H_ME-DS10776_2-DS10776_2
               H_ME-DS10776_3-DS10776_3 ... H_ME-DS11313_3-DS11313_3
               H_ME-DS11313_1-DS11313_1
colData names(1): id
pedigree(4139): famid id fid mid sex dx
complete trios(320):
```

Create a GRanges object such that each range contain 64 rare variants.

```
[1] 64
```

```
64
```

```
99
```

```
> window <- rowData(ste)[1:10] + 1000
> system.time(scan.trio <- ScanTrio(object = ste.rare,
  window = window, block = range(rowData(ste.rare))))
```

```
window: 1
window: 2
window: 3
window: 4
window: 5
window: 6
window: 7
window: 8
window: 9
window: 10
      user system elapsed
3.844    0.024    3.885
```

```
> scan.trio
```

DataFrame with 10 rows and 7 columns

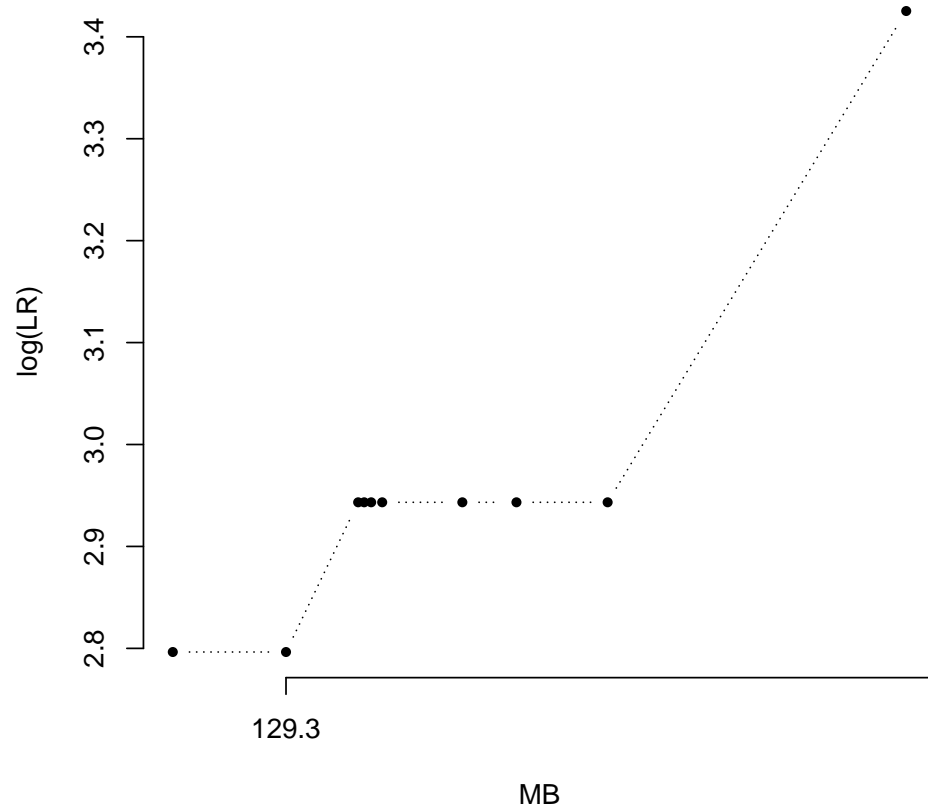
	lr	minor.in	major.in	minor.out	major.out	mendel.in
	<numeric>	<integer>	<integer>	<integer>	<integer>	<integer>
1	16.38618	11	3	5437	6011	0
2	16.38618	11	3	5437	6011	0
3	18.97866	13	4	5435	6010	0

4	18.97866	13	4	5435	6010	0
5	18.97866	13	4	5435	6010	0
6	18.97866	13	4	5435	6010	0
7	18.97866	13	4	5435	6010	0
8	18.97866	13	4	5435	6010	0
9	18.97866	13	4	5435	6010	0
10	30.73175	14	4	5434	6010	0

mendel.out

<integer>

1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0



5.2 Holger-style Genotype Matrix

Coercion to matrix for *trio*

```
> geno <- as(ste, "matrix")
> aTDT(geno[, 1:5])
```

```
$stat
[1] 16.13333 26.47059 21.55172 20.16129 35.00000
```

```
$pval
[1] 5.903578e-05 2.675813e-07 3.444129e-06 7.117887e-06 3.297053e-09
```

```
$transMinor
[1] 4 2 2 3 0
```

```
$transMajor
[1] 26 32 27 28 35
```

Or apply the aTDT method to the SNP Trio Experiment directly.

```
> aTDT(ste[1:5])
```

```
$stat
[1] 23.405063 3.368421 26.888889 27.000000 5.254545
```

```
$pval
[1] 1.312327e-06 6.645742e-02 2.154942e-07 2.034555e-07 2.188952e-02
```

```
$transMinor
[1] 18 30 14 15 19
```

```
$transMajor
[1] 61 46 58 60 36
```

A Count of Transmission of Variants (*TransCount()*)

A.1 Window

```
> window <- gr[100] + 10000
> window2 <- gr[100] + 1000
> block <- range(rowData(ste))

> TransCount(ste, window)
```

A.2 Not in window, but in block

```
> TransCount(ste, setdiff(block, window))
```

A.3 In the whole block

```
> TransCount(ste, block)
```

A.4 In both windows

```
> TransCount(ste, GRangesList(window, window2))
```

A.5 Not “In both windows,” but in block

```
> TransCount(ste, GRangesList(setdiff(block, window), setdiff(block,
  window2)))
```

B Count of Transmission of Rare Variants (*TransCount()*)

B.1 In both windows

B.2 Not “In both windows,” but in block

C Classes

C.1 SnpMatrix, DataFrame, GRanges

The four key ingredients are the SNP matrix, the pedigree information as a DataFrame, position of the SNPs given by a GRanges object, and covariate data given as a DataFrame.

```
> sm
> ped.DF
> gr
```

C.2 SummarizedExperiment

We combine three of the key ingredients when we create the SummarizedExperiment object.

```
> se <- SummarizedExperiment(assays = SimpleList(geno = t(sm)),
  colData = col.DF, rowData = gr)
```

C.3 SNPTrioExperiment

Now, we include the pedigree information as an object of class `PedClass`. We keep `PedClass` independent of `SNPTrioExperiment` for flexibility.

```
> ped <- PedClass(ped.DF)
> ste <- SNPTrioExperiment(se, pedigree = ped)
```

Here is the show method.

```
> ste
```

And now we verify that it is indeed an extension of `SummarizedExperiment`.

```
> getClass("SNPTrioExperiment")
```

C.4 PedClass

Now we investigate the pedigree slot of the `SNPTrioExperiment` object.

```
> class(pedigree(ste))
> getClass("PedClass")
> pedigree(ste)
```

C.5 geno accessor

```
> class(geno(ste))
> getClass("SnpMatrix")
> geno(ste)
```

C.6 RowData

```
> rowData(ste)
```

C.7 ColData

D Validity