

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

Samuel G. Younkin

March 16, 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 FamilyExperiment

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

```
class: FamilyExperiment
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: FamilyExperiment
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):
```

```
> window <- rowData(ste.rare) + 250000
> system.time(scan.trio <- ScanTrio(object = ste.rare,
  window = window, block = range(rowData(ste.rare))))
```

```
      user  system elapsed
18502.82    17.41  18522.39
```

```
> scan.trio
```

DataFrame with 6397 rows and 7 columns

	lr	minor.in	major.in	minor.out	major.out	mendel.in
	<numeric>	<integer>	<integer>	<integer>	<integer>	<integer>
1	2.453997	1478	1699	3970	4315	0
2	2.453997	1478	1699	3970	4315	0
3	2.453997	1478	1699	3970	4315	0
4	2.453997	1478	1699	3970	4315	0
5	2.453997	1478	1699	3970	4315	0
6	2.453997	1478	1699	3970	4315	0
7	2.129824	1483	1699	3965	4315	0
8	2.129824	1483	1699	3965	4315	0
9	2.182580	1483	1700	3965	4314	0
...	...	...	...	...	...	...
6389	1.622246	1270	1449	4178	4565	0
6390	1.589757	1270	1448	4178	4566	0
6391	1.572905	1266	1443	4182	4571	0
6392	1.485017	1266	1440	4182	4574	0
6393	1.464183	1264	1437	4184	4577	0
6394	1.581660	1255	1431	4193	4583	0
6395	1.585349	1254	1430	4194	4584	0
6396	1.625565	1252	1429	4196	4585	0
6397	1.625565	1252	1429	4196	4585	0

mendel.out

```

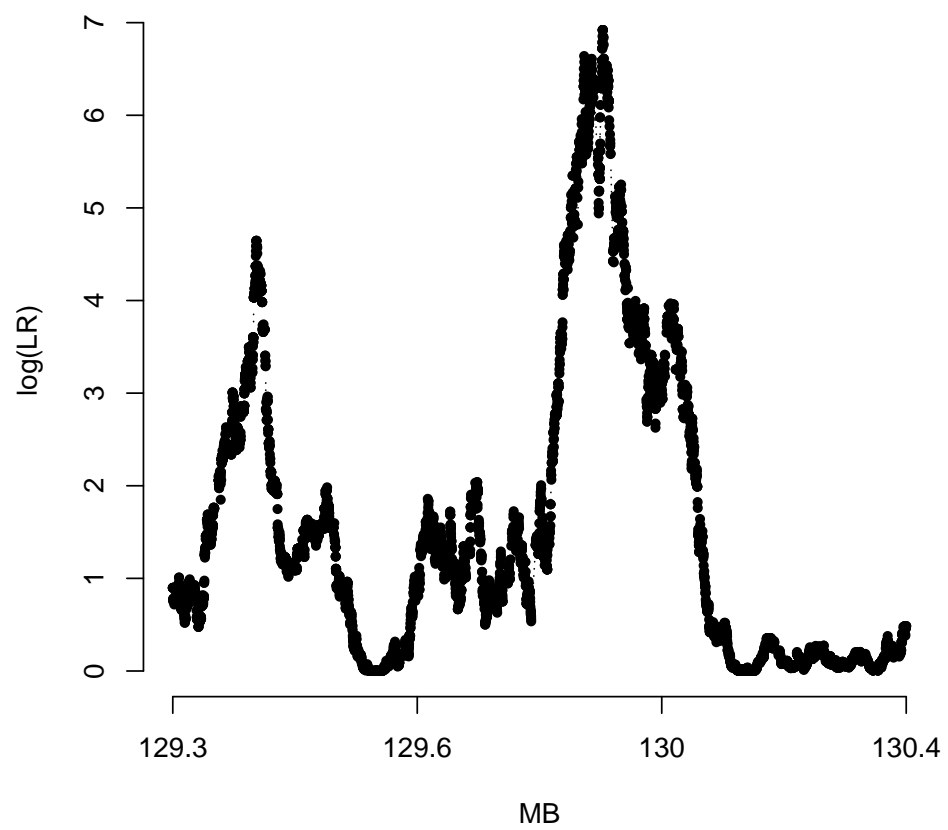
      <integer>
1          0
2          0
3          0
4          0
5          0
6          0
7          0
8          0
9          0
...      ...
6389      0
6390      0
6391      0
6392      0
6393      0
6394      0
6395      0
6396      0
6397      0

```

```

> save(scan.trio, file = "../data/scan-trio.RData")

```



## 5.2 Holger-style Genotype Matrix

Coercion to matrix for *trio*

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

Or apply the aTDT method to the FamilyExperiment directly.

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

## 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 FamilyExperiment

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

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

Here is the show method.

```
> ste
```

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

```
> getClass("FamilyExperiment")
```

### C.4 PedClass

Now we investigate the pedigree slot of the `FamilyExperiment` 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