Prioritizing GWAS Results and Identifying Risk SNP-Associated Functional Annotation Tree with ‘**GPATree**’ Package

Aastha Khatiwada[1], Bethany J. Wolf[1], Ayse Selen Yilmaz[2], Paula S. Ramos[1,3], Maciej Pietrzak[2], Andrew Lawson[1], Kelly J. Hunt[1], Hang J. Kim[4], Dongjun Chung[2]

[1]Department of Public Health Sciences, Medical University of South Carolina, Charleston, South Carolina, USA
[2]Department of Biomedical Informatics, The Ohio State University, Columbus, Ohio, USA
[3]Department of Medicine, Medical University of South Carolina, Charleston, South Carolina, USA
[4]Divison of Statistics and Data Science, University of Cincinnati, Cincinnati, Ohio, USA

02/28/2021

1 Overview

This vignette provides an introduction to the `GPATree` package. R package `GPATree` implements GPA-Tree, a novel statistical approach to prioritize genome-wide association studies (GWAS) results while simultaneously identifying the combinations of functional annotations associated with risk-associated genetic variants. GPA-Tree integrates GWAS summary statistics and functional annotation data within a unified framework, by combining a decision tree algorithm (CART)(Leo et al. 1984) within the hierarchical model.

The package can be loaded with the command:

```
> library(GPATree)
```

This vignette is organized as follows. Sections 2.1 and 2.2 illustrate the recommended `GPATree-ShinyGPATree` workflow, which provides convenient and interactive genetic data analysis interface. Advanced users might also find Sections 2.3.1 – 2.3.3 useful as the command lines can be used for integrating GPA-Tree as part of the more comprehensive genetic data analysis workflow, for example.

Please feel free to contact Dongjun Chung at `chung.911@osu.edu` for any questions or suggestions regarding the ‘`GPATree`’ package.

2 Workflow

In this vignette, we illustrate the GPA-Tree analysis workflow, using the simulated data provided as the `GPATreeExampleData` in the `GPATree` package. In the simulated data, the number of SNPs is set to $M = 10,000$ and the number of functional annotations is set to $K = 10$. The GWAS association p-values and the binary functional annotation information are stored in `GPATreeExampleData$gwasPval` and `GPATreeExampleData$annMat`, respectively. The number of rows in `GPATreeExampleData$gwasPval` is assumed to be the same as the number of rows in GPATreeExampleData$annMat, where the i-th ($i = 1, ..., M$) row of gwasPval and annMat correspond to the same SNP.

```
> data(GPATreeExampleData)
> dim(GPATreeExampleData$gwasPval)
[1] 10000     1
> head(GPATreeExampleData$gwasPval)
          P1
SNP_1 0.7454
SNP_2 0.4894
SNP_3 0.6026
SNP_4 0.1496
SNP_5 0.2538
SNP_6 0.3161
> dim(GPATreeExampleData$annMat)
[1] 10000    10
> head(GPATreeExampleData$annMat)
      A1 A2 A3 A4 A5 A6 A7 A8 A9 A10
SNP_1  1  0  0  0  0  1  0  0  0   1
SNP_2  1  0  0  0  0  0  0  0  0   0
SNP_3  1  0  0  0  0  0  0  0  0   1
SNP_4  1  0  0  0  0  0  0  0  0   0
SNP_5  1  0  0  0  1  1  0  0  0   0
SNP_6  1  0  0  0  0  1  0  0  0   0
```

2.1 Fitting the GPA-Tree Model

We can fit the GPA-Tree model using the GWAS association p-values (GPATreeExampleData$gwasPval) and functional annotation data (GPATreeExampleData$annMat) described above, using the code shown below.

```
> fit.GPATree <- GPATree(gwasPval = GPATreeExampleData$gwasPval,
+                        annMat = GPATreeExampleData$annMat,
+                        initAlpha = 0.1,
+                        cpTry = 0.005)
```

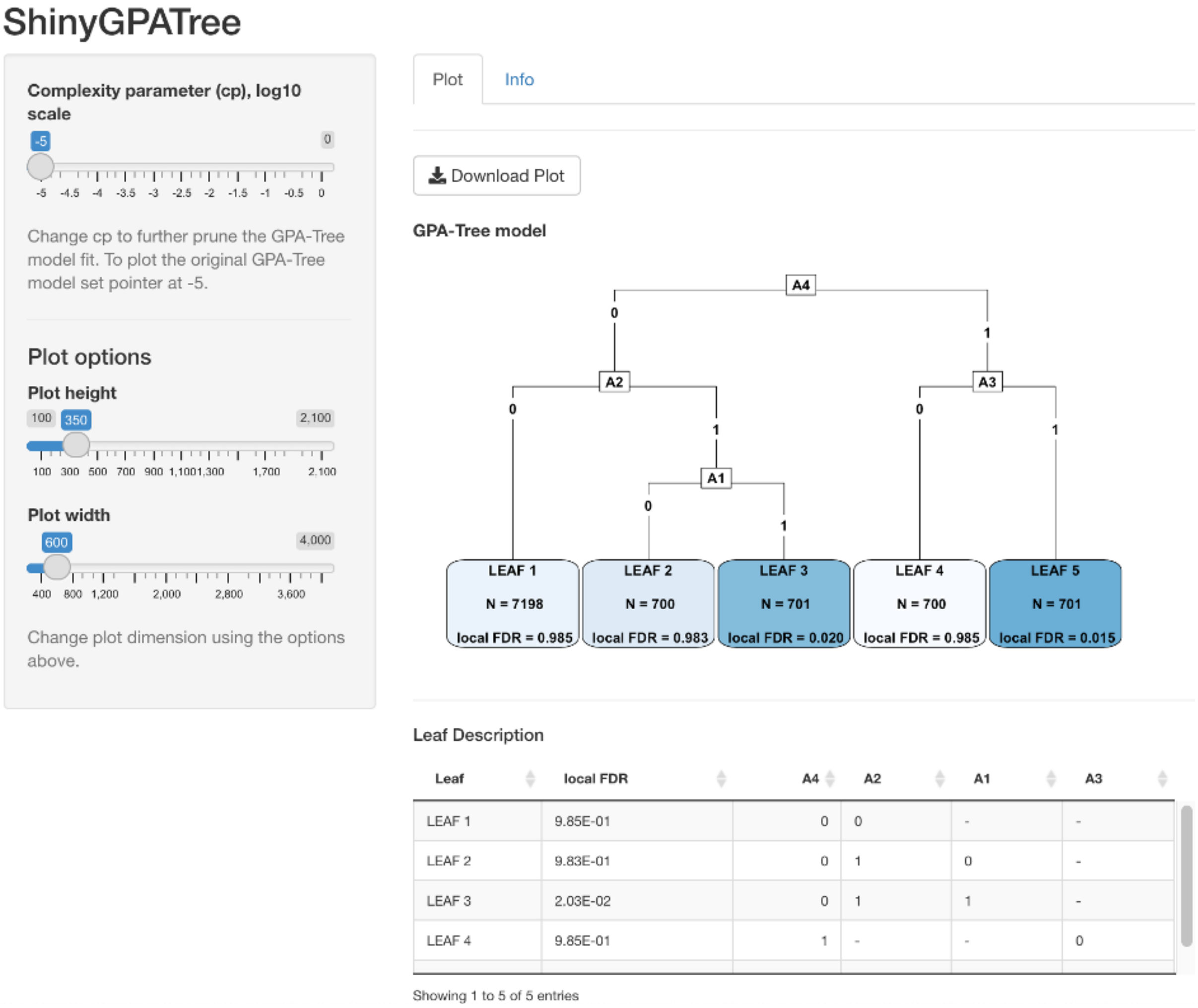
```
> fit.GPATree
Summary: GPATree model results (class: GPATree)
--------------------------------------------------
Data summary:
    Number of GWAS data: 1
    Number of Annotations: 10
    Number of SNPs: 10000
    Alpha estimate: 0.4999
Functional annotation tree description:
       local FDR A4 A2 A1 A3
LEAF 1    0.9849  0  0  -  -
LEAF 2    0.9834  0  1  0  -
LEAF 3    0.0203  0  1  1  -
LEAF 4    0.9850  1  -  -  0
LEAF 5    0.0154  1  -  -  1
--------------------------------------------------
```

2.2 ShinyGPATree

The following command can be used to initialize the ShinyGPATree app. ShinyGPATree allows for interactive and dynamic investigation of disease-risk-associated SNPs and functional annotation trees using R Shiny.

```
> ShinyGPATree(fit.GPATree)
```

Figure 1 shows the layout of the ShinyGPATree app, where the 'Plot' tab opens by default. The summary statistics displayed in the plot are automatically updated as the user input option for cp (in the log10 scale) on the left panel of the screen is modified. Users can also improve visualization of the functional annotation tree plot using the plot width and height options on the left panel. The 'Download Plot' button on the top allows users to download the functional annotation tree plot as a Portable Network Graphics (png) format file. Finally, a table titled 'Leaf Description' underneath the plot characterizes the functional annotations that are 0 or 1 for SNPs in specific leaves.



Complexity parameter (cp), log10 scale

-5 … 0

Change cp to further prune the GPA-Tree model fit. To plot the original GPA-Tree model set pointer at -5.

Plot options

Plot height

100 350 … 2,100

Plot width

600 … 4,000

Change plot dimension using the options above.

GPA-Tree model

Leaf Description

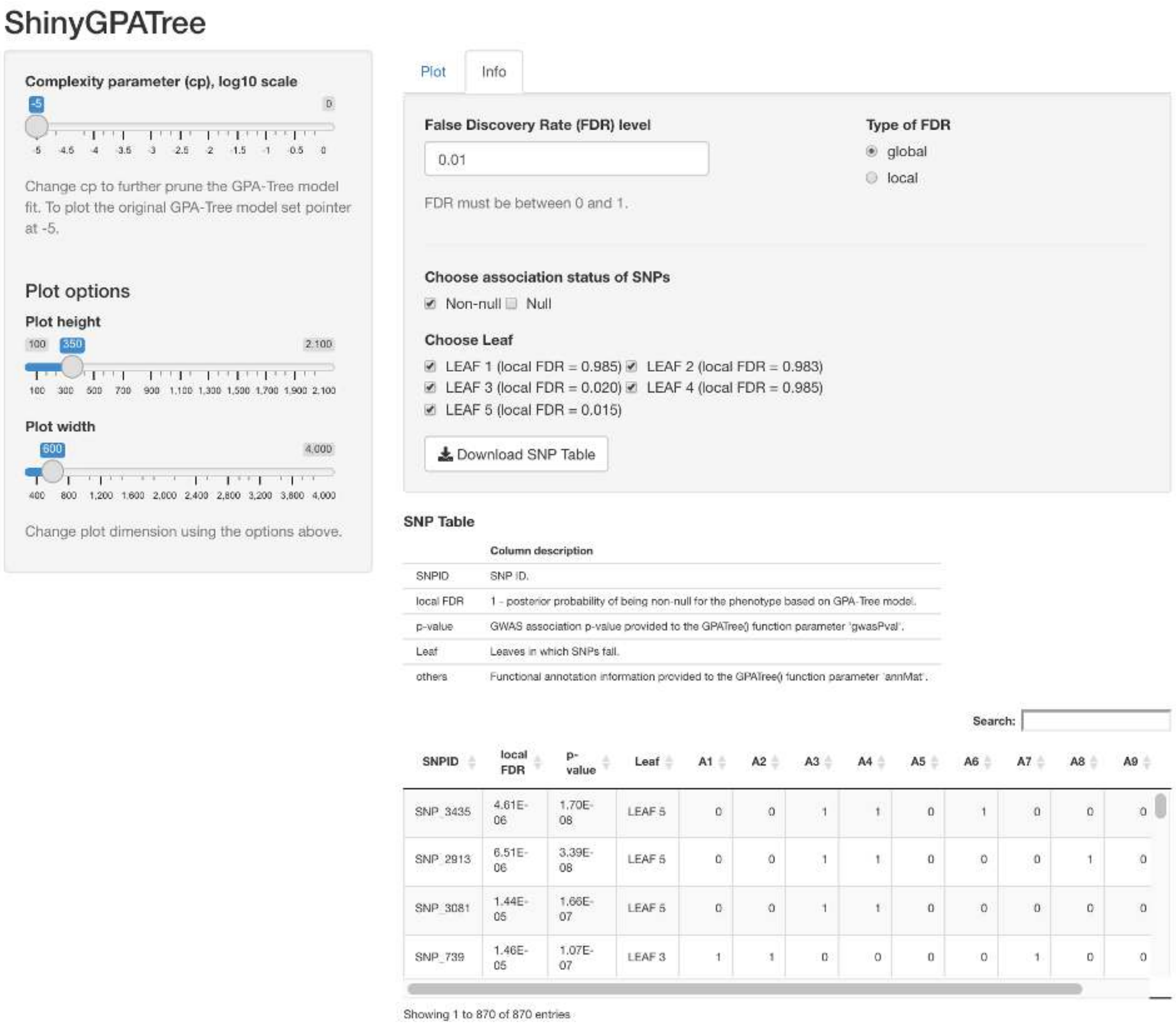
Leaf	local FDR	A4	A2	A1	A3
LEAF 1	9.85E-01	0	0	-	-
LEAF 2	9.83E-01	0	1	0	-
LEAF 3	2.03E-02	0	1	1	-
LEAF 4	9.85E-01	1	-	-	0

Showing 1 to 5 of 5 entries

Figure 1: Screenshot of the ShinyGPATree app with the 'Plot' tab open.

As seen in Figure 2, the 'Info' tab in the ShinyGPATree app opens the user interface for association mapping and functional annotation characterization for SNPs. Under this tab, users can find more information on specific SNPs driving the visualization. At the top of the panel, user input options for FDR level and FDR type (global vs. local) are located, followed by options to select SNPs that fall on specific leaves of the

GPA-Tree model or have specific association status (non-risk-associated vs. risk-associated SNPs). The 'SNP Table' at the bottom of the 'Info' tab panel shows information for SNPs that satisfy all user-specified input options. Each row of the table represents a SNP and includes its ID, local FDR value, GWAS association p-value, the leaf in which it is located, and its complete functional annotation information. The 'Download SNP Table' button allows users to download the 'SNP Table' as a Microsoft Excel comma separated values (CSV) file format.



Complexity parameter (cp), log10 scale: -5

Change cp to further prune the GPA-Tree model fit. To plot the original GPA-Tree model set pointer at -5.

Plot options

Plot height: 350

Plot width: 600

Change plot dimension using the options above.

False Discovery Rate (FDR) level: 0.01

FDR must be between 0 and 1.

Type of FDR

● global
○ local

Choose association status of SNPs

☑ Non-null ☐ Null

Choose Leaf

☑ LEAF 1 (local FDR = 0.985) ☑ LEAF 2 (local FDR = 0.983)
☑ LEAF 3 (local FDR = 0.020) ☑ LEAF 4 (local FDR = 0.985)
☑ LEAF 5 (local FDR = 0.015)

SNP Table

	Column description
SNPID	SNP ID.
local FDR	1 - posterior probability of being non-null for the phenotype based on GPA-Tree model.
p-value	GWAS association p-value provided to the GPATree() function parameter 'gwasPval'.
Leaf	Leaves in which SNPs fall.
others	Functional annotation information provided to the GPATree() function parameter 'annMat'.

SNPID	local FDR	p-value	Leaf	A1	A2	A3	A4	A5	A6	A7	A8	A9
SNP_3435	4.61E-06	1.70E-08	LEAF 5	0	0	1	1	0	1	0	0	0
SNP_2913	6.51E-06	3.39E-08	LEAF 5	0	0	1	1	0	0	0	1	0
SNP_3081	1.44E-05	1.66E-07	LEAF 5	0	0	1	1	0	0	0	0	0
SNP_739	1.46E-05	1.07E-07	LEAF 3	1	1	0	0	0	0	1	0	0

Showing 1 to 870 of 870 entries

Figure 2: Screenshot of the ShinyGPATree app with the 'Info' tab open.

2.3 Advanced use

2.3.1 Prunning GPA-Tree model fit

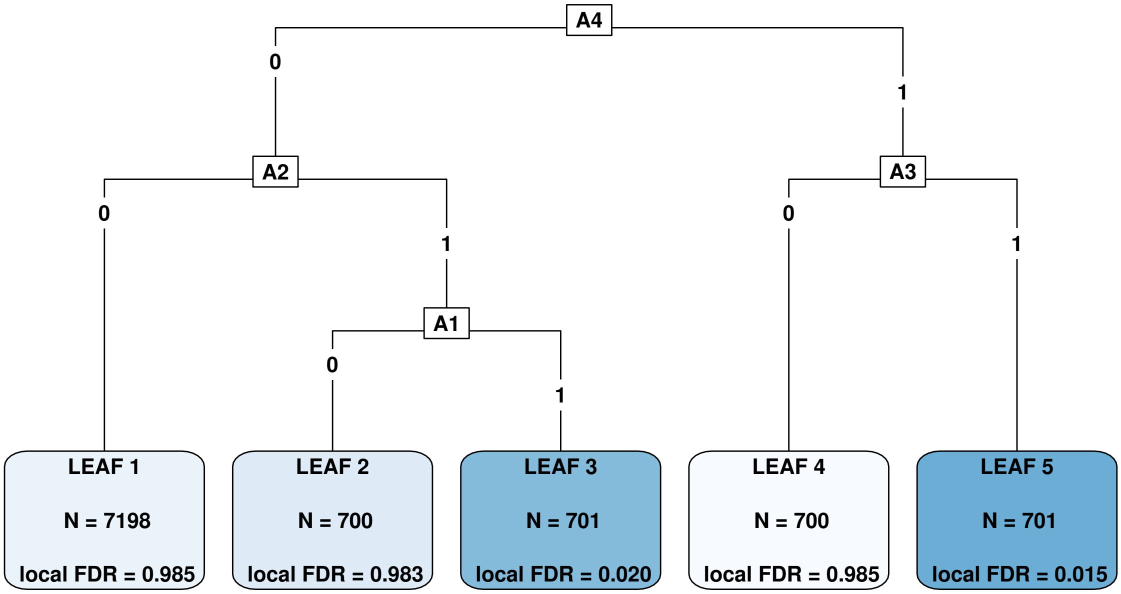
The `prune()` function will prune the GPA-Tree model using any cp value between 0 and 1 as shown below.

```
> fit.GPATree.pruned <- prune(fit.GPATree, cp = 0.3)
> fit.GPATree.pruned
Summary: GPATree model results (class: GPATree)
--------------------------------------------------
Data summary:
    Number of GWAS data: 1
    Number of Annotations: 10
    Number of SNPs: 10000
    Alpha estimate: 0.4999
Functional annotation tree description:
       local FDR                    Note
LEAF 1    0.8492 No annotations selected
--------------------------------------------------
```

2.3.2 Functional annotation tree

The plot() and leaf() functions will plot the GPA-Tree functional annotation tree and provide information about the leaves (terminal nodes) in the tree as shown below.

```
> plot(fit.GPATree)
```



```
> leaf(fit.GPATree)
       local FDR A4 A2 A1 A3
LEAF 1    0.9849  0  0  -  -
LEAF 2    0.9834  0  1  0  -
LEAF 3    0.0203  0  1  1  -
LEAF 4    0.9850  1  -  -  0
LEAF 5    0.0154  1  -  -  1
```

2.3.3 Association mapping

For the fitted GPA-Tree model, we can make inferences about SNPs using the assoc() function by: (1) prioritizing risk-associated SNPs, and (2) identifying the leaves of the GPA-Tree model in which the risk-associated SNPs are located. The assoc() function returns two columns. The first column contains binary values where 1 indicates that the SNP is associated with the trait and 0 indicates otherwise. The second column provides information regarding the leaf in which the SNP is located in the GPA-Tree plot. The assoc() function allows both local (fdrControl="local") and global FDR controls (fdrControl="global") and users can set the threshold to be between 0 and 1 using the 'FDR' argument. For GPATreeExampleData, GPA-Tree model identified 870 risk SNPs at the nominal global FDR level of 0.01. 371 and 499 of the 870 risk-associated SNPs are located in leaf 3 and leaf 5, respectively. The following lines of code can be used to investigate association mapping and functional annotation tree.

```
> assoc.SNP.GPATree <- assoc(fit.GPATree,
+                            FDR = 0.01,
+                            fdrControl="global")
> head(assoc.SNP.GPATree)
      P1   leaf
SNP_1  0 LEAF 1
SNP_2  0 LEAF 1
SNP_3  0 LEAF 1
SNP_4  0 LEAF 1
SNP_5  0 LEAF 1
SNP_6  0 LEAF 1
> table(assoc.SNP.GPATree$P1)

   0    1
9130  870
> table(assoc.SNP.GPATree$leaf)

LEAF 1 LEAF 2 LEAF 3 LEAF 4 LEAF 5
  7198    700    701    700    701
> table(assoc.SNP.GPATree$P1, assoc.SNP.GPATree$leaf)

    LEAF 1 LEAF 2 LEAF 3 LEAF 4 LEAF 5
  0   7198    700    330    700    202
  1      0      0    371      0    499
```

References

Leo, Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. *Classification and regression trees*. CRC press.