GSEA.plot

Civelek Lab University of Virginia

Sarah Innis, Warren Anderson, Mete Civelek

March 6,2019

This vignette gives an introduction into the GSEA.plot package. This package builds off of the Gene Set Enrichment computational analysis published by the Broad Institute specifically their R code for outputting the results. This package uses the enrichment data to create plots that are easer to understand and modify.

This package requires three data files which can come from any folder that the path must be clearly defined. It will output two summary results files and a pdf of all of the enrichment plots to the working directory specified in wd__new. Within that directory there should be a results folder named with the same string as the doc.string input, this is where the individual gene set results will go.

Load Data and Package

```
library(devtools)
#install_github("Sarah-innis/GSEA.plot")
library(GSEA.plot)

# data locations
code.folder="C:/Users/Student/Documents/CivelekLab/GSEA.plot/R/"
datasets.folder="C:/Users/Student/Documents/CivelekLab/GSEA.plot/data/"
genesets.folder="C:/Users/Student/Documents/CivelekLab/GSEA.plot/data/"

#file for outputs-- must contain Leukemia_hall folder inside it
wd_new="C:/Users/Student/Documents/CivelekLab/GSEA_edit3/"
```

GSEA Analysis

The GSEA analysis takes three files. The expression data in gct format. The phenotype for each sample in cls format and a gene set database in gmt format. Other parameters include the folder locations of these files, thresholds for analysis of the significance of the data, and options to control how the graphs look.

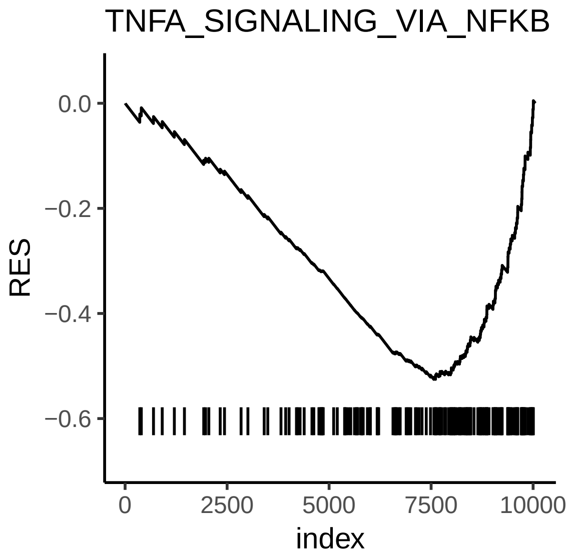
```
 pp= GSEAplots(wd_new=wd_new, datasets.folder=datasets.folder,input.ds.name="Leukemia.gct",
               input.cls.name="Leukemia.cls", genesets.folder=genesets.folder,
               gene.set.input="h.all.v6.1.symbols.gmt", code.folder=code.folder,
               doc.string="Leukemia_hall", nperm=1000,nom.p.val.threshold=-1,fdr.q.val.threshold = 0.25
               bar_percent=0.1, gap_percent=0.1, under_percent=0.1,upper_percent=0.1,color_line="black"
               color_tick="black")
```

Function outputs

Plots

This is the enrichment plot given from the first geneset.

```
pp$plots[[1]]
```



Use the function plot.ES to save a pdf containing the enrichment plots for every set in the gene set database.

```
plot.ES(list.of.plots=pp$plots,plotname="Vignette_example")
```

Gene.Set.Reference.Matrix and Gene.set.leading

These are the gene symbols contained in the first gene set.

```
names(pp$gene.set.reference.matrix)[[1]]
#> [1] "HALLMARK_TNFA_SIGNALING_VIA_NFKB"
head(pp$gene.set.reference.matrix[[1]])
#> [1] "JUNB"    "CXCL2"   "ATF3"    "NFKBIA"  "TNFAIP3" "PTGS2"
```

These are the leading gene symbols for the first gene set.

```
names(pp$gene.set.reference.matrix)[[1]]
#> [1] "HALLMARK_TNFA_SIGNALING_VIA_NFKB"
head(pp$gene.set.leading[[1]])
#> [1] "CEBPB" "MCL1"  "PLAU"  "IL18"  "PLEK"  "BCL3"
```

Reports

One report is generated for each phenotype. The table shows all of the information available in the report file except for the source for the geneset which is ommitted for easier viewing.

```
kable(pp$report1[1,-3], format="latex", booktabs=TRUE) %>%
  kable_styling(latex_options="scale_down")
```

GS	SIZE	ES	NES	NOM p-val	FDR q-val	FWER p-val	Tag %	Gene %	Signal	FDR (median)	glob.p.val
HALLMARK_WNT_BETA_CATENIN_SIGNALING	28	0.53627	1.8905	0	0.082816	0.055	0.357	0.131	0.311	0	0.032

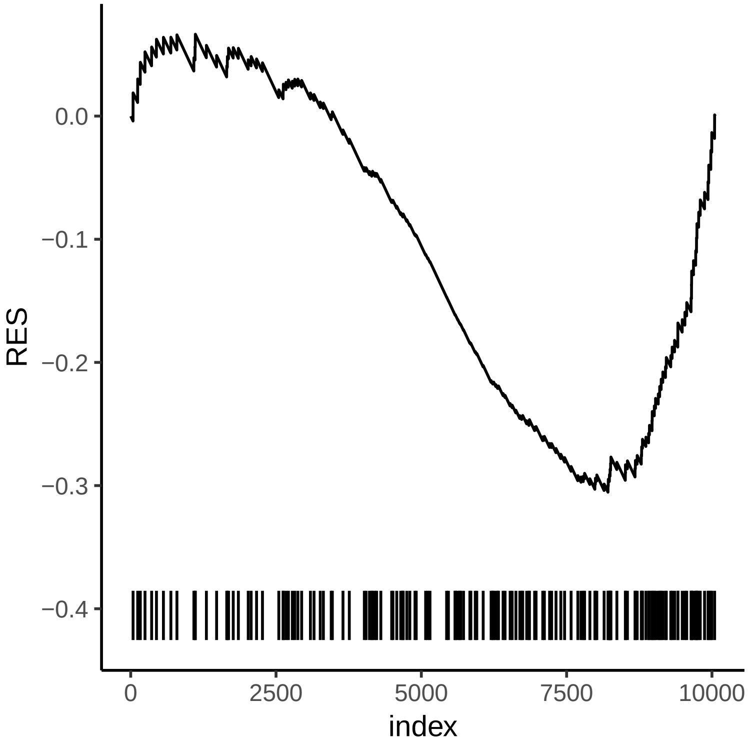
Enrichment Scores

The enrichment score ouput allows you to access the data used to generate the plots so that you might create your own.For example if you wanted to quickly compare the enrichment scores between two gene sets.

```
data1=pp$ES[[2]]
kable(head(data1),format="latex",booktabs=TRUE)
```

index	RES	EStag
1	-0.0001011	0
2	-0.0002021	0
3	-0.0003032	0
4	-0.0004043	0
5	-0.0005054	0
6	-0.0006064	0

```r
enrich_ind=which(data1$EStag==1)
d=data.frame(x=enrich_ind, y=matrix(min(data1$RES)-0.12,length(enrich_ind),1), vx=matrix(0,length(enric
p <- ggplot(data1, aes(index,RES))+geom_line(col="black")
p <- p+geom_segment(data=d, mapping=aes(x=x, y=y, xend=x+vx, yend=y+vy),col="black")
p <- p+theme_classic()
print(p)
```


RES
index